CALIGARI 2

MANUAL

OCTREE SOFTWARE, INC.

Caligari 2

Modeling and Animation

by Roman Ormandy

Copyright 1991 by Octree Software

Table of Contents

1. Introduction 4

1A. What is Caligari? 6

1B. Hardware requirements 8

1C. Instalation 9

1D. Menu Structure and Mouse Operation . . 11

2. Object Design 13

2A. Base Menu 17

2B. Navigation 19

2B(1). Viewpoint Manipulation . . . 23

2B(2). Object Manipulation 32

2B(3). Views 41

2B(4). 3D Grid, Erase and Undo . . . 46

2B(5). Keyboard Shortcuts 49

2C. Status 51

2D. Tools 54

2D(1). Glue/Unglue/Copy/Mirror . . . 55

2D(2). Extruder 62

2E. Single Point Editing 73

2F. Load/Save 81

2G. Primitives 85

2H. Render 87

2H(1). Color 91

2I. NUM 95

3. Scene Composition 101

3A. Base Menu 104

3B. Tools 106

3B(1). Lights 109

3C. AddObj/File 111

3D. QRender 114

3D(1). Color 116

4. Animation 121

4A. Interactive animation 122

4B. Script based animation 129

1

Introduction

Our world has three dimensions. Every day we perform a multitude of physical actions requiring complex coordination in 3D without a moments hesitation. Yet, using the existing software for 3D manipulation and animation is usually so difficult that it requires hard thought and, often, special training. Few designers and artists are willing to undergo this arduous learning process.

Because 3D visualization is essential in many areas of human activity and communication, Octree Software firmly believes that the future of personal computing must inevitably be three dimensional. To fulfill this promise, innovative hardware and software must be created to perform these tasks.

Caligari is a new kind of a software tool. It enables the user to think directly in 3D space instead of transferring designs from paper. **Caligari** also allows the user to move and change objects in real time so that the user can undertake very complicated projects and see them through to completion in a fraction of the time needed heretofore.

Caligari is designed for everyone who's tasks require **3D visualization.** This would include corporate presentations, industrial design, interior and architectural design, training, education and fine art. For those with the highest demands, a professional version of **Caligari** is available.

Octree Software believes that **Caligari** will establish a new market for 3D conceptual design and animation. For those who have waited for a sophisticated and affordable tool, the wait is over.

1A

What is Caligari?

Caligari is a conceptual three-dimensional design and animation program for the Amiga family of computers. Its user interface introduces a new paradigm for human interaction with a computer in a 3D environment.

Caligari represents the third generation in user interface design. The first generation, exemplified in IBM PC - and MS DOS - based systems, is character based using the keyboard as the only means of entering the information into the computer. The user enters symbolic strings from the command line using the keyboard. This type of interface is essentially one dimensional and based on **SYMBOLS** (characters).

The second generation interface is represented by the Apple Macintosh and its operating system. It is based on two dimensional **ICONS** rather than symbols. The user interacts with the computer screen by moving the mouse cursor in two dimensions and selecting icons to activate program functions. This interface is also characterized by movable "pull-down" or "pop-up" windows which the user can drag about the screen and scale in size to see several windows at once.

The third generation interface uses a far more effective paradigm:

1. **Caligari** interfaces the user with **OBJECTS**. The user sees actual representations of real world physical objects with 3D spatial and physical attributes.

2. The user's actions are performed in **virtual** three-dimensional space. This space is a close approximation of the physical world of our everyday experience.

3. The response is "**real time**", that is as the user moves, rotates, scales and modifies his 3D objects by moving the mouse cursor, the screen image instantly reacts to show these changes.

4. In **Caligari** the motion of the user's hand manipulating virtual 3D objects on the screen imitates the motion of a hand manipulating similar objects in the real world. Therefore, no learning period is necessary for using the system since the user does not need to master any new skills.

5. **Caligari** does not incorporate a window-based environment. Rather than looking at the window, the user literally looks through the window into a convincing 3D environment. It is a direct representation of the 3D world, one in which the user can perform the same actions as in everyday life.

6. **Caligari**'s objects can be manipulated **hierarchically**. That is, not only whole objects but also any part of an object can be selected and manipulated assuming the structure of the object allows for it. In **Caligari**, hierarchical access is fully transparent to the user. For example, if the user wants to manipulate a robot's arm, he can select it and then rotate it around the robot's shoulder joint. This happens automatically without any need for the user to specify the structure or the level of hierarchy.

Because it would be impossible to fully describe here the dynamic interaction that occurs between the user and the computer, we refer to the videotape tutorial accompanying this manual, which will provide in visual terms the exciting experience of **Caligari**'s interface.

1.B Hardware requirements.

Caligari2 will run on any Amiga model with a minimum of 2 Mb or memory. Hard disk is strongly recommended.

Caligari2 will take advantage of a 68030 CPU with a floating point accelerator. Consequently, these CPU's are also strongly recommended.

The Amiga PC is a powerful graphics computer with advanced user interface and lots of computing power devoted to graphics. It does not allow, however, for full-color display output, a capability indispensable for high-end video applications.

Fortunately, there are available full color add-on frame buffers which improve the quality of the Amiga output, provided that software will take advantage of their presence.

Caligari supports a number of popular frame buffers such as HAME from Black Belt Systems and DCTV from Digital Creations. For more about the supported frame buffers, see the documentation on the program disk.

For software compatibility Raster Link from Active Circuits and Art Department from ASDG are recommended if the user desires to convert images between various existing formats. On 3D side Caligari will read and write Videoscape, Lightwave and Sculpt 3D data format.

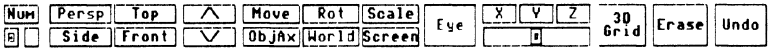
1.C Instalation

We refer to the instalation instructions supplied with the program disk, since actual steps may change slightly with various revisions of the software. Please read them carefully and follow them step by step.

1D

Menu structure and mouse operation

The first thing to note about the menus is their location. **Caligari**'s menus are located at the bottom of the screen resembling the instrument panel of a car or airplane. They are comprised of horizontal strips. Rather than hierarchical, they are **stackable**, i.e. they stack one on top of the other as the user activates them. Using stackable menus, the user can create a personalized working environment. Any selection from any visible menu strip can be made instantly without going up or down the hierarchy. When the user exits an environment such as the object module, he will find it unchanged when he returns.



Selections from the menus are made by moving the mouse cursor on top of the desired item and clicking the left mouse button. This highlights the menu item and activates the function the item will perform.

When the cursor crosses the top menu strip into the 3D workspace the cursor changes its shape. This indicates that it now relates to objects in the workspace rather than in the menu items. If a mouse button is depressed while the cursor is in the work area, the cursor will disappear and the currently selected function will begin to operate. When the button is released, the function will stop operating and the cursor will appear again.

While the menu strips automatically stack themselves on top of the menu when activated, they can be individually eliminated by selecting the "close" button at the lower left side of the menu strip. The menus can be brought up again by selecting the appropriate choice from the base menu. In this way, the user can set up the structure of menus best suited to the particular task at hand.

All of the menus can be made invisible by selecting the close button on the base (bottom) menu. Now the whole screen becomes a 3D workspace, with the current function active. Thus the user can position an object anywhere on the screen with a completely uncluttered view.

IMPORTANT!

The menus can be brought back by depressing the right mouse button.

As an alternative the user can activate various menu options as well as toggle the menu on/off through the keyboard shortcuts. Please see the section 2B(5).

The default working configuration of the program is saved in a configuration file. When the user changes a parameter, such as the number of segments for extrusion, he can save the current configuration so that the next time the program is loaded it will reflect the saved changes.

Caligari file menus are context sensitive. For example, LOAD menu will display only available objects when activated in object design and available polygons when called from the extruder. Additionally, the standard file requester enables the user to see a whole directory as well as to change the current directory.

2.

OBJECT DESIGN

The user enters this module immediately after the title page. **Caligari** opens into the Object Design module with the base menu and the navigation menu present at the bottom of the screen. "Rotate" "Eye" in "World" space and the X-Y-Z coordinate axes are selected. The selection of a function is indicated by an amber highlight of that function.

The introductory screen represents a perspective "Through the Window" view of an empty three dimensional environment. The user can identify his location in this 3D environment thanks to the dark groundplane (grid) also seen in the perspective view.

The user can move anywhere in this 3D world simply by selecting "Move" and moving the mouse on the surface of the desk while holding the left mouse button down. As the mouse moves, the 3D groundplane changes accordingly, creating a smooth movement in this 3D world. This eye-hand feedback facilitates a remarkably fluid coordination of user actions in 3D world.

Two user motions are possible in the 3D world. The first "Rotates" the users point of interest (selected object), enabling him to see the "world" from different angles. The second option is "Move", which lets the user move in the world linearly. No matter which action the user performs and regardless of the complexity of the world, **Caligari** always provides the user with a real time response to his actions. This response is achieved by **Caligari**'s ability to make a precise interpretation of the user's actions and display an instantaneous reaction to them.

In addition to movement in the world, the user also has an option to manipulate the 3D objects in it in various ways. A simple way of creating an object in the 3D world is to use the "Primitives" menu. Here, a collection of basic 3D shapes such as "cube", "cone", "sphere" and "cylinder" are presented visually on the screen. The user can choose any one of the shapes and it will immediately appear on the perspective grid in the worldspace. Once an object is

placed in this world it can be manipulated by the user with real time ease.

On the enclosed video tape we see how a half-cylinder is being moved, rotated and scaled simply by moving the mouse cursor. This is unique to **Caligari** and its interpretation of the user's actions, which is so natural that very little time is needed to learn the system. An object can be copied and combined with one or more other objects to create a highly complex object. The user can combine two objects by selecting the "glue" operation and simply pointing to the another object to be glued to the selected object. Once the two objects are glued they behave as one object, i.e., they move, rotate, etc., together. While objects appear on the screen as wireframes for real time manipulation, they can also at any point be rendered as solids with **Caligari**.

The **solid view** is obtained by selecting the "Render" from the base menu. The selected object immediately appears as a solid, i.e. with hidden surfaces eliminated and various light intensities present due to a lightsources added to the scene. When in solid view, the objects cannot be moved. They can, however, be interactively colored. This can be accomplished by selecting a new color and pointing to the object.

Another unique aspect of **Caligari**'s design is the way it handles complex objects. Not only can the user move the object as a whole, he can also manipulate any part of a complex object with the same ease as in real life. This is called "hierarchical access" and is demonstrated on the enclosed video tape by swinging the robot's arm around its shoulder joint. While other programs allow for hierarchical manipulation, the simplicity of **Caligari** is unique. The user need not specify anything about the object structure other than how the parts are to be glued together. After that he merely points to the desired subobject (arm) and it automatically performs the correct motions.

In addition to perspective view, the user can choose orthogonal views of the world (top, side, front). In orthogonal view the user can position different objects with great precision due to the program's ability to zoom, in real time, on any particular part of the design. There are almost no numerical entries necessary.

If a regular placement of an object is needed, the user simply selects the 3D grid option. Objects will now jump from position to position (as specified by the user) instead of in a smooth motion.

In addition to combining primitive shapes to create a new complex object, the user can draw complex shapes by hand. This is accomplished in the "**extruder**". With the extruder the user can draw a shape by hand, place a number of holes inside it (such as a curve inside the letter A) and then sweep this 2D shape through the 3D space to create a 3D volume. Again, very little experimenting is necessary to use this feature due to the ergonomic design of the extruder.

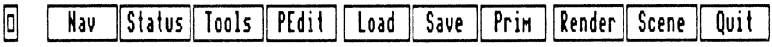
One of the most powerful Caligari features is "Single Point Editing" ("**PEdit**"). It allows the user to select a single point, edge, face or part of an object and move it, scale it or rotate it. The user can also extrude a face or part of an object. The objects can be mirrored or sliced through with a plane.

Finally, the objects created can be saved for later recall and use in the Scene Composition module of **Caligari**.

2A

BASE MENU

The base menu is always present at the bottom of the screen both in Object Design and Scene Composition and contains buttons for activation of major functions of the program. It can be made invisible (along with the other menus) by selecting its "close" button. It can be made visible again by clicling the right mouse button.



These are the major functional blocks in Caligari. When selected, some of them activate additional menu strips, some replace the existing menu setup, and some open a completely new screen.

NAV - Navigation is the central part of **Caligari**'s interface. It enables the user to move in the 3D environment and manipulate objects in it.

STATUS - Displays numerical data relating to the current scene.

TOOLS - Contains GLUE, UNGLUE, EXTRUDE, COPY and MIRROR.

PEDIT - Allows for manipulation of point, edge, face or part of an object.

LOAD - Displays menu for loading objects into the workspace.

SAVE - Permits saving objects under user-selected names.

PRIM - Displays iconic menu of primitive (basic) objects for instant loading into the workspace.

RENDER - Displays current object as a solid, in color. This is a Quick Render.

SCENE - Exits Object Design and enters Scene Composition module.

QUIT - Exits the program.

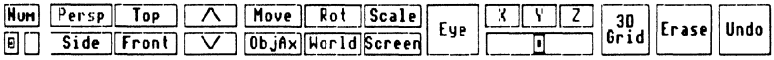
2B.

NAVIGATION

The navigation menu is by far the most often used menu in **Caligari**. When the user enters the Object Design, the navigation menu is the default selection for the workspace.

In the navigation menu the user can move around in the 3D design workspace and manipulate the objects in it. Because the user sees the perspective plane in the workspace, there is never any confusion as to which action is currently being performed.

The height of the navigation menu is equal to two text lines.



On the left side of the navigation menu are options for view selection. In the middle are operations for object and eye manipulations (Move, Rot, Scale), X, Y, and Z axis indicators and Object/World/Screen coordinates toggle. On the the right side are the **3D Grid Undo** and the **Erase** button.

Most user actions are performed directly in the perspective view ("**Persp**"). For precise positioning the user can elect to display the workspace in three orthogonal views, "**Front**", "**Side**" and "**Top**". Regardless of the projection, most functions operate identically in each view.

"/\ \/" enables the user to select an active subobject. "**Move**", "**Rot**" and "**Scale**" are operations on current object or the viewpoint. "**X**", "**Y**" and "**Z**" describe the parameters of the currently selected object or eye manipulation function.

"**ObjAx/World/Screen**", indicates which coordinate space the current operation operates in. Combination of four views, three coordinate spaces and three geometric transformation on either View or current object allows for rich maneuverability and manipulability of objects.

"**Eye/Obj**" allows the user to select whether the current operation is performed on the selected object or on the viewpoint.

The Object coordinate system. If an object consist of more simple objects (see glue and unglue), each simple object, each subpart of complex object and whole object have their own axes and a coordinate system connected with them.

The Screen coordinate system is associated with the screen of a display. The "X" axis is horizontal from left to right, the "Y" axis is vertical and the view into the screen creates the "Z" axis. The center of this coordinate system is in the eye.

"**3D Grid**" toggles smooth or step motion for an object or a viewpoint. "**Erase**" removes a selected object from the workspace; "**Undo**" undoes last performed operation.

The navigation menu works identically in the Object Design and the Scene Composition module. Many functions can be performed from the keyboard using the NUM submenu. This submenu also enables 3D grids for translation, rotation and scaling of objects.

Axis selection.

For each operation a set of axes must be specified. Only along these axes the manipulation operation will be affected. Because a mouse provides only two degrees of freedom, all three axes can be controlled by mouse only if proportional (same for x, y, z) transformation is required. The user can do this by pressing both mouse buttons simultaneously. If only left button is pressed, a transformation is affected along X and Y axes (if they are specified by gadgets).

Right button affect only Z axis. Axis selection is therefore a combination of the selected button and the mouse buttons depressed.

Object selection.

Caligari uses dynamic selection of objects. When a mouse is moved in the workspace and a button (or buttons) is depressed, the object under the cursor is immediately selected as active and currently selected operation will be performed on it. This allows the user to easy manipulate any object in a scene. If more than one object is under the cursor, the smallest one is selected. If no object is under the cursor when a button is depressed, the previously selected object is kept.

The selected object will be highlighted immediately after its selection and the previously selected object will be unhighlighted (draw in the background color).

In Caligari, the user can create the hierarchy by gluing various parts of the complex object together (see glue and unglue). For example, with the simple robot, first the hand and forearm are glued together, then the new object is glued to the upper arm. The whole arm is then glued to the robot's shoulder. At any time, any part of the robot's body can be accessed for manipulation.

The "Up arrow","Down arrow" gadgets and dynamic pick facility of mouse allows to select any part of complex object. Clicking on the "Down" gadget, while a complex object is selected first selects subobject of this object. Selection of any other subobjects at the same level is easy with dynamic pick, i.e. by moving a cursor on top of the desired subobject and depressing a mouse button. Each of these subobjects can also be a complex object and sequence of "Down" and dynamic pick can continue as far as some simple object will be selected. Clicking on "Up" gadget selects next higher level. If any subobject of a complex object is selected, it will appear light bluish-green, while the rest of the object is orange.

Dynamic Pick can be disabled from the NUM menu. If it is disabled, the objects can be selected by moving the cursor on top of the desired object and **double clicking on the RIGHT button.** This is sometimes very useful, particularly in the Scene Composition, where many objects can be simultaneously on the screen.

It is necessary to use Save option from NUM menu to make change of settings for Dynamic Pick permanent. This setting can be different for the Object Design and the Scene Composition.

2B(1)

VIEWPOINT MANIPULATION

The viewpoint motion is the movement of the user in the environment. In **Caligari**, the user movement can be readily discerned from the object manipulation even if there is only one (or none) object in the workspace. The viewpoint motion can take place in the object, world or screen space.

The World coordinate system is associated with the global 3D space the user looks at. You can imagine it as three orthogonal vectors radiating from the center of the groundplane defining the workspace. X points down one axis of the ground plane, Y points across the other axis and Z points up through the groundplane. If the user selects, for example, movement along only the X axis the viewpoint will move parallel to one side of the grid and this will not change even if the user rotates the view in a different direction.

The Object coordinate system is local to the object itself. If the object is rotated, then its coordinate system will be rotated with it. In this way, for example, the robot's local X axis will always be perpendicular to the front plane of the robot, regardless of the robot's position in the world.

The Screen coordinate system is associated with the viewpoint itself. Since the user looks at the world through the screen, it is also associated with the screen. Good thing about this coordinate system is that the up is always up (Move Y) and forward is always forward (Move Z).

There are several options on the navigation menu that allow the user to change his or her position in the 3D world. "**Move**", which translates the viewpoint in X,Y and Z in the selected coordinate system; "**Rot**" in World or Object coordinate system, which orbits the viewpoint around the World center or the active object; and "**Rot**" in the Screen coordinate system, which changes the viewing angle relative to the ground plane without moving the viewpoint just like pan or tilt. Also, the user can change the focal length of eye with the "**Scale**" button. In all those cases "**Eye**" button must be selected too.

One of the strengths of **Caligari** is that the user is always aware of his position in the world and of the changes in this world. Movement of the viewpoint is clearly indicated by the perspective change of the ground plane. While the viewpoint is moving, only the groundplane and the selected object (if any) are visible with the rest of the scene removed during the action. The scene will be rerendered in the background color (green) once the user lifts the finger from the mouse button.

The most important thing to remember is that when the viewpoint changes its position, the whole world (and every object in it) changes its appearance.

Purpose:

To move the viewpoint linearly in the worldspace.

Procedure:

1. If the Move function does not appear on the menu, select the Nav function and the Move function will appear.

2. Select the Eye and the Move function from menu.

3. Select the desired coordinate system and the axes for movement by clicking on ObjAx,World,Screen and X,Y,Z.

4. Move the mouse into the workspace, hold down the left or the right buttonwhile moving the viewpoint location accordingly.

Options:

ObjAx - move in the active Object coordinate system

World - move in the World coordinate system

Screen - move in the Screen coordinate system

X - move along x axis Y - move along y axis Z - move along z axis

Note:

Left mouse button controls x and y axes, right button controls z axis. The viewpoint is always moved in the same direction as the mouse, i.e. to move the eye to the left, the mouse must be moved to the left too. Only if the desired direction is perpendicular to the screen, the mouse must be moved sideways.

If Object coordinate system is selected to move the eye in, first the mouse cursor must be moved above required object before mouse buttons will press to select the object. If no object is under cursor while button is pressed, the

coordinate system of previously active object (highlighted) will be used.

Example:

- Select Eye, Move, Screen and make sure that all three axes (X,Y,Z) are selected. If there are any objects in the workspace, erase them.

- Press the right mouse button and pull the mouse towards you. The viewpoint will "fly" away from the ground plane.

- Press the left button while holding the right button down, then release the right button. Move the mouse away and towards you and you will "fly" up and down. Move the mouse sideways and you will "fly" sideways.

Eye Rot (World/ObjAx)

Purpose:

To encircle a point you are looking at by moving the viewpoint on the surface of an imaginary sphere.

Procedure:

1. Select the Eye and the Rot function from menu and select the World or Object coordinate system.

2. Move the mouse into the workspace, hold down the left or the right buttonwhile moving the mouse to encircle the object or world center.

Options:

ObjAx - turn the viewpoint around the active object

World - turn the viewpoint around the World center

X - rotate around world horizon - turn along meridian of imaginary - sphere

Y - rotate around world z axis - turn along parallel of imaginary sphere

Z - rotate around screen z axis - turn sideways

Note:

Left mouse button controls x and y axes, right button controls z axis.

The viewpoint is always turn in the same direction as the mouse, i.e. to turn the eye to the left, the mouse must be moved to the left too. Only if the desired axis is perpendicular to the screen, the mouse must be moved sideways.

Example:

- Load a cone and a cylinder into the workspace (using the Prim menu). Make sure that there are not other objects in the workspace. Select Eye, Rot with ObjAx.

- Press the left mouse button and move the mouse sideways. You will be encircling the two shapes. Notice that the selected object is the center of the circular motion.

- Do 180-degree turn. You will now be looking at the scene from the opposite direction.

- Select Move/Obj/World and translate one of the objects. Notice that even though you are on the opposite side of the scene the object still follows the cursor.

Eye Rot (Screen)

Purpose:

To change the angle of the eye relative to the ground plane. The eye can look up or down, left or right, or roll clockwise or counterclockwise.

Procedure:

1.If the function does not appear on the menu, select the Nav function and the Rot function will appear.

2.Select the Eye and the Rot function from menu and select the Screen coordinate system.

2.Move the mouse into the workspace, hold down the left or the right buttonwhile moving the mouse to change the viewpoint angle accordingly.

Options:

X - Tilt up or down the world horizon

Y - Pan left or right the world vertical

Z - turn sideways (bank)

Note:

Left mouse button controls x and y axes, right button controls z axis.

In this operation the position of the viewpoint does not change.

The viewpoint is always turned in the same direction as the mouse, i.e. to turn the eye to the left, the mouse must be moved to the left too. Only if the z axis is desired, the mouse must be moved sideways.

Purpose:

To change the focal length of the eye. It can be adjusted from "Tele-lens" to "Fish-eye".

Procedure:

1.Select the Eye and the Scale function from menu.

2.Move the mouse into the workspace, hold down the left mouse button to change the focal length of the eye lens.

Note:

The coordinate system and axes are not relevant for this operation.

The horizontal slider also gives a visual indication of the focal length setting. Scale Eye is different from the Move command in that the actual viewpoint does not change.

In the top, left and front view the Scale/Eye (together with Move) makes possible a precise positioning of objects. The user can move around the two-dimensional view of the world and magnify the area of interest to an extremely high degree. Once the detail is magnified, precise positioning of objects is easy.

Example:

- Load a few objects into the workspace. Make sure you are in the perspective view.
- Drag the slider located next to the Eye button all the way to the left. The scene will become much smaller.
- Select Move, press the right button (making sure the Z is on) and move the mouse away from you. The scene will come closer with the effect of an extreme wide angle of the camera.

2B(2)

OBJECT MANIPULATION

When manipulating objects, the viewpoint remains stationary.

Object manipulation consist of movement, rotation or scaling of a wireframe object. Two objects may be "glued" together to create a new compound object.

First an object must be selected. It can then be manipulated in real time with the mouse. For each operation the coordinate system and a set of axes must be specified. For example, if X, Y and Z are set for a scaling operation, the object will be scaled proportionally in all three dimensions. If X and Y axes are turned off, then the scaling operation will affect only the Z axis and the object will be stretched or squeezed along this axis, the dimensions along X and Y remaining unchanged.

There are three options for setting the coordinate system in which Object Manipulation commands operate: World coordinate system (global), Object coordinate system (local) and Screen coordinate system.

The World coordinate system is associated with the global 3D space the user looks at. You can imagine it as three orthogonal vectors radiating from the center of the groundplane defining the workspace. X points down one axis of the ground plane, Y points across the other axis and Z points up through the groundplane. If the user selects, for example, translation along only the X axis the object will move parallel to one side of the grid and this will not change even if the user rotates the object in a different direction.

The Object coordinate system is local to the object itself. If the object is rotated, then its coordinate system will be rotated with it. In this way, for example, the robot's local X axis will always be perpendicular to the front plane of the robot, regardless of the robot's position in the world.

The Screen coordinate system is associated with the viewpoint itself. Since the user looks at the world through the screen, it is also associated with the screen. One reassuring thing about this coordinate system is that up is always up (Move Y) and forward is always forward (Move Z). The Object coordinate system forms the basis for the hierarchical manipulation capability of **Caligari**. The

user creates the hierarchy by gluing various parts of the complex object together. For example, with the simple robot, first the hand and forearm are glued together, then the new object is glued to the upper arm. The whole arm is then glued to the robot's shoulder. At any time any part of the robot's body can be manipulated, and the manipulation will be performed in any space since each subobject has its own unique coordinate system which can be defined with "**Axes**". button on the PEdit menu (see Single Point Editing).

Thus, if the user selects the robot's arm and then Rot Z, the arm will rotate around the robot's shoulder. This can be overridden, however, by selecting the world or screen space.

Example:

- Load the object "robot" from the disk.
- Select down arrow (\/). The robot's head will appear light bluish-green, the rest of the robot's body is orange. This is to indicate that the head of the robot can now be manipulated.
- Point to the robot's body and depress the left mouse button. The body will now be highlighted.
- Select the down arrow again and click on the robot's arm. It will become highlighted.
- Select "Rot" from the navigation menu. Make sure that "ObjAx" and "Z" is highlighted.
- Move the cursor to the workspace area, press the right mouse button and move the mouse. The arm of the robot will swing around its shoulder "joint". Release the button.
- Select down arrow again and click on the robot's forearm. Move the cursor to the workspace area, press the right mouse button and move the mouse.
- The robot's forearm will swing around its elbow "joint".

To select the active object, the "**Dynamic Pick**" is used:

Anytime the mouse button is pressed in the design space, an object under the cursor will be selected for manipulation. If no object is under cursor, previously active one will be used. If an active object is a complex object, any subpart of it can be selected using the "Down arrow", "Up arrow" and the "Dynamic Pick" facility.

Dynamic Pick can be turned off if the need arises. This will happen for example in the scene module with a lot of objects. To turn the Dynamic Pick off the user can unselect Dynamic Pick from the NUM menu.

Of course objects can be selected even if Dynamic Pick is turned off. To pick an object the user will position the cursor on top of it and **double click** the RIGHT button.

By default, dynamic Pick is enabled in Object Design and disabled in Scene Composition. Clicking on the Save button in the NUM menu allows the user to change the default.

Purpose:

To move the object in an arbitrary direction.

Procedure:

1.Select the the Move function and make sure the OBJ is selected.

2.Select the desired coordinate system and the axes for movement by clicking on ObjAx,World,Screen and X,Y,Z.

4.Move the mouse into the workspace above the object you want to move and hold down the left or the right button to move this object.

Options:

ObjAx - move in active object coordinate system

World - move in World coordinate system

Screen - move in Screen coordinate system

X - move along x axis Y - move along y axis Z - move along z axis

Note:

Left mouse button controls x and y axes, right button controls z axis. The object is always moved in the same direction as the mouse, i.e. to move the object to the left, the mouse must be moved to the left too. Only if the desired direction is perpendicular to the screen, the mouse must be moved sideways.

Example:

- Load the "cone" from the Primitives menu. Select "Move" and X, Y from the Navigation menu. Make sure that "World" is highlighted.
- Move the mouse cursor to the workspace area. Press the left button and move the mouse on the surface of the desk.
- The "cone" will follow the mouse movements on the surface of the perspective grid.

Purpose:

To rotate the object around an arbitrary axis or axes.

Procedure:

1.Select the the Rot function and make sure the OBJ is selected.

2.Select the desired coordinate system and the axes for rotation by clicking on ObjAx,World,Screen and X,Y,Z.

4.Move the mouse into the workspace above the object you want to rotate and hold down the left or the right button to rotate this object.

Options:

ObjAx - rotate around active object axes

World - rotate around World axes

Screen - rotate around Screen axes

X- rotate around x axis Y- rotate around y axis Z - rotate around z axis

Note:

An object is rotated always around its center, regardless of which coordinate system is selected.

Left mouse button controls x and y axes, right button controls z axis. Since the left button controls two axes simultaneously, it may be useful sometimes to turn X or Y off and position object more precisely. Simultaneous control of two axes is most effective in Screen or World space.

The object always turns in the same direction as the mouse is moved, i.e. to turn the object to the left, the mouse must be moved to the left too.

Only if the desired axes is perpendicular to the screen, the mouse must be moved sideways.

Example:

- Load the half cylinder from the Primitives menu. Select Rot and make sure Obj and X,Y,Z are highlighted. Move the mouse cursor to the workspace, press the right button and move the mouse again. The object will rotate along its vertical axis.

- Rotate the object slightly sideways by pressing the left mouse button and moving the mouse in the workspace.

- Press the right mouse button again and move the mouse in the workspace area and notice that the object rotates along its vertical axis even though it is now positioned differently.

Scale / Object

Purpose:

To change the size or shape of an object.

Procedure:

1.Select the the Scale function and make sure the OBJ is selected.

2.Select the desired coordinate system and the axes for scaling by clickingon ObjAx,World,Screen and X,Y,Z.

4.Move the mouse into the workspace above the object you want to scale and hold down the left or the right button or both to scale this object.

Options:

ObjAx - scale in active object coordinate system

World - scale in World coordinate system

Screen - scale in screen coordinate system

X - scale along x axis Y - scale along y axis Z - scale along z axis

Note:

Left mouse button controls x and y axes, right button controls z axis.

Both buttons control proportionally all specified axes.

If only the left or right button is pressed, the object is scaled along the direction of mouse motion. Only if desired axis is perpendicular to the screen, the mouse must be moved sideways. If both buttons are pressed, the mouse must always be moved sideways.

Scaling in the world or the screen space will deform the shape of an object unless it is aligned with the world or screen axes.

Example:

- Load a cylinder from the Primitives menu. Select Scale making sure that X, Y and Z are highlighted as well as Obj.

- Scale the cylinder up to make it larger by pressing both the left and right buttons and moving the mouse. Now, scale the cylinder down by only pressing the right mouse button and moving the mouse. Notice that this time only the thickness of the cylinder changed, not its diameter.

2B(3)

VIEWS

Caligari can display both Perspective and Orthogonal projections. In both cases it provides real time response and the interface is identical.

THE PERSPECTIVE VIEW is natural for the user because it closely resembles the world as we see it in everyday life. A characteristic property of perspective projection is that objects appear smaller as they get further from the viewpoint.

THE ORTHOGONAL VIEW does not perform perspective division, consequently objects do not change size as they move away. This view is good for very precise positioning of objects. There are three orthogonal views: **Top, Side and Front.**

Caligari automatically disables operations which cannot be perceived properly in a two dimensional orthogonal view. For example, when looking from above at the scene (Top view) up and down motion of objects (Z axis) is disabled.

Example:

- Load a cylinder and a cone from the Primitives menu. Both objects will initially occupy the same space.
- Select "Side" to obtain the side view. Select "Move" and "Eye" and pan around the side view so that both objects are in the center of the screen.
- Select "Move", select "Obj" and slide the cone on the top of cylinder.
- Select "Scale", "Eye" and magnify the objects until they fill the whole screen. Select "Move" again (unselect Eye) and position the cone very precisely on top of the cylinder.
- Select "Top", then "Side" to check that the cone is directly aligned with the top of the cylinder. Select "Persp" to go back to the perspective view.

Persp

Purpose:

To display a three dimensional perspective view of the world.

Procedure:

1. Make sure you are not in the Perspective view.
2. Select "Persp" from the Navigation menu.
3. Immediately, the world will be shown in perspective.

Note:

Perspective resets to its default view after exiting from the Object Design module.

Example:

- Select Persp for the perspective view if you are not already in it. Load a cube from the Primitives menu.
- Select "Move", "Obj", X and Y and make sure that World is selected. Move the cube about the perspective groundplane and observe that the size and image of the cube changes as it moves.

Top

Purpose:

To display the world from above (in the X-Y plane).

Procedure:

1. Make sure you are not in the Top view

2. Select the "Top" button from the Navigation menu.

3. Immediately, the world will be shown from above.

Note:

In the Top view Rot/Eye is disabled. Also Move Z is disabled.

Top view (together with other orthogonal views) is well suited for precise positioning of design elements. This can be accomplished by zooming close to the current object and doing fine adjustments for rotation and translation of the object.

Example:

- Select Top and load a cube from the Primitives menu. Notice that the cube appears as a square.

- Select "Move", "Obj", X and Y and move the cube about the screen. Notice that neither the size nor the shape of the cube changes.

Side

Purpose:

To display the world from the side (in Z-Y plane).

Procedure:

1. Make sure you are not in the side view.

2. Select the "Side" button from the Navigation menu.

3. Immediately, the world will be shown from the side.

Note:

In the Side view, Rotate/Eye are disabled. Also Move X is disabled.

Side view (together with other orthogonal views) is well suited for precise positioning of design elements. This can be accomplished by zooming close to the current object and doing fine adjustments for rotation and for translation (positioning) of the object.

Front

Purpose:

To display the world from the front (in X-Z plane).

Procedure:

1. Make sure you are not in the front view

2. Select the "Front" button from the Navigation menu.

3. Immediately, the world will be shown from the front.

Note:

In the Front view Rotate/Eye are disabled. Also Move Y is disabled.

Front view (together with other orthogonal views) is well suited for precise positioning of design elements. This can be accomplished by zooming close to the current object and doing fine adjustments for rotation and for translation of the object.

Purpose:

To erase the currently selected (highlighted) object.

Procedure

1. Select the object to be erased.

2. Select the "Erase" function. The object will disappear from the workspace.

Note:

Erase removes an object without prompting for confirmation and the erased object cannot be retrieved. Erase only works on complete objects, a subobject can not be erased.

Erase only removes objects from the workspace. Previously saved objects on the disk are still available.

In addition to erasing objects, Erase will clear the currently selected edge-set in the Single Point Editing. Be careful, because after clearing the selected vertices, the next click on erase will erase the object itself.

Purpose:

To undo the previously executed function.

Procedure:

1. After incorrectly editing or altering the dimensions of an object, select the "Undo" function.

2. Immediately the last change made will be undone.

Note:

The operations which can be undone are Move, Rot, Scale, as well as Slice.

Only the last performed action can be undone. However, the last Undo can also be undone.

The fact that the last geometrical operation can be undone is useful when experimenting with a conceptual design. At any time a part of the design can be modified from an existing to a new composition by modifying the part while the mouse button is depressed. When an acceptable position is found and the button is released the new composition can be evaluated. If it is still not satisfactory, selection of "Undo" will return to the old composition, ready for further modifications.

[Esc] key can also be used to Undo.

Move

The grid can be turned on selectively with a free motion in one or more axes and a step motion in the other axes. The translation grid is a very powerful tool which facilitates creation of large regular structures in a matter of seconds.

Example:

- Load the wedge from the Primitives menu. Select 3DGrid. Make sure that Move/World/Obj is also selected. Now move the mouse cursor. The wedge will move in a step motion. Select Tools and Copy. Move the mouse cursor again. A second wedge will step in a position just next to the first. Repeat Copy and move for the third wedge. You could now glue the three wedges together and copy them with the grid still on.

Rot

Grid can be turned on selectively to provide free motion in one or more axes and step motion in the other axes. Grid Rot allows fast rotation of an object in regular increments (45-90 degrees).

Example:

- Load the cylinder from the Primitives menu into the workspace. Select 3D Grid.
- Now select Rot X from the Navigation menu and move the mouse cursor. The cylinder will rotate in 22.5 degree increments. Make four steps and the cylinder will be now perfectly aligned horizontally.

Scale

Purpose:

To select step value for object scaling.

2B(5) Keyboard Shortcuts

Caligari's user interface design is based on direct manipulation of objects using the mouse and the keyboard was not originaly intended as an input devices except for naming the files. However, many experienced users asked for "hot keys" for the most often used operations activated by the left hand which would enable them to switch the operation without the need to do multiple selection by the mouse from the menu.

So here they are:

Q - Move **W** - Rotate . .**E** - Scale

A - Object Axes **S** - World Axes **D** - Screen Axes

Z - X **X** - Y . . . .**C** - Z

[Space Bar] - Object/Eye "Toggle"

left [Shift] - Menu On/Off "Toggle"

[Esc] - Undo

Keyboard shortcuts only work when the Navigation menu is active. Additionally, there are a few key combinations allowing for easier editing of string fields, used in some menus in Caligari:

[Shift][Del] or **[right Amiga][x]** - erase current string field

[Tab] or **[down arrow]** - go to the next field

[Shift][Tab] or **[up arrow]** - go to the previous field

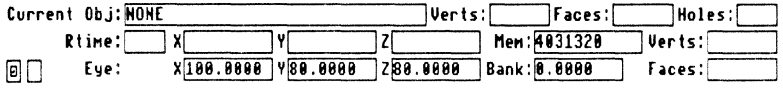
[Shift][left arrow] - go to the beginning of the field

[Shift][right arrow] - go to the end of the field

2C

STATUS

When "Status" is selected from the Base menu, a new menu strip will appear just above the existing menu bars and will display and dynamically update the numerical information relating to the Viewpoint and the currently selected object.



The status window is passive, that is, the user can not numerically modify the values by using the keyboard. For numerical entry the "**NUM**" menu must be selected. These are the Status variables:

Current obj - The name of the currently selected object will be displayed in this field. If a subobject is selected, the name of the subobject will be displayed. The user CAN select objects by typing their names into this field and pressing Enter key.

RTime - Time necessary to render the current object in wireframe mode. Time is displayed in sixtieths of a second.

X, Y, Z - Coordinates of the current object in the worldspace. When the object is moved, the coordinates are immediately updated.

Vertices- number of vertices of the current object. (also for the whole scene)

Faces - Number of faces of the current object (also for the whole scene)

Holes - Number of holes for the current object

Mem - Amount of available memory.

Eye X, Y, Z - Coordinates of the current viewpoint position.

Bank - Rotation of the viewpoint in the screen space. (0 = horizontal)

The Status menu gives the exact numerical location of both the objects and the viewpoint.

2D

TOOLS

In the Navigation menu the objects can be positioned and shaped using Move, Rot and Scale. The "Tools" menu, on the other hand, offers a selection of tools to create complex objects out of simpler ones. This menu also offers a way to draw shapes with the mouse.

The Tools menu works in conjunction with the Navigation menu. It simply stacks itself above any existing menus.

Glue/Unglue is a basic tool for combining simpler objects together. For example, several objects can be combined together to form an arm. Two arms can then be glued to the torso of a robot. **Caligari** remembers exactly how the structure of a complex object (such as a robot) was created. This provides the basis for hierarchical manipulation of complex objects.

The **Extruder** is actually a **collection** of very powerful tools. With it, the user can draw any 2D shape by hand and sweep it through 3D space to create a 3D object.

Copy replicates objects. Note that the new copy of the object is in the same location as the old copy of the object. The user must translate, scale or rotate the new copy of the object, before the new and the old copies of the object will appear as separate.

Mirror will create a mirrored copy of an object. Since there is only one topologically different mirror, the mirroring plane does not have to be specified. Mirror is quite powerful, the user can create for example half of a persons face, copy it and mirror the copy. The mirrored copy can be then glued to original to create a full head.

Negate tool will reverse the direction of surface normals for every polygon of selected object.

Decompose will separate an object which is not hierarchical but consist of disjunct sets of faces into simple subobjects connected into hierarchy.

Purpose:

To glue one object to another object. Any number of objects may be glued together but they must be glued together one at a time.

Procedure:

1. Select first (root) of the objects to be glued together. The object appears highlighted.

2. If the "Glue" option does not appear on the menu, select the Tools function and it will appear.

3. Select Glue from the Tools menu. The rest of the Tools menu disappears.

4. Select a second object to glue to the first object. Both objects will appear highlighted and are then combined into one object. The Tools menu reappears.

Note:

Glue option is available only in the Object Design module. The order in which the objects are glued together is remembered for hierarchical access. See also, "ObjAx". If the user decides not to glue after "Glue" was selected, a second click on the same button or on any other button will bring the Tools menu back.

Example:

- Load a cylinder and a cube from the Primitives menu. Select "Scale" Z and flatten the cube to about a quarter of its original thickness. Make sure that the Tools menu is up.

- Select Glue and point to the cylinder. Press the left mouse button and both objects will be glued together. Select "Render" to see the resulting object as a solid.

Unglue

Purpose:

To separate two objects previously glued together.

Procedure:

1. If "Unglue" does not appear on the menu, select the Tools function and it will appear.

2. Select an object which is composed of several glued objects.

4. Using the Down arrow, select the subobject to be separated from the main object. It is highlighted in a bluish hue. The rest of the complex object is orange.

3. Select Unglue from the menu. The subobject will then be highlighted in white, while the rest of the complex object will have background color. The highlighted subobject is now a separate object.

Note:

The Unglue command is available only in the Object Design module. Its main use is reuse of previously designed parts in new constructs. This enables rapid creation of new designs out of old ones.

Copy

Purpose:

To create a replica of an object.

Procedure:

1. If "Copy" does not appear on the menu, select the Tools function and it will appear.

2. Select an object.

3. Select Copy from the Tools menu.

4. The newly created object occupies the same space as the original object.

Note:

Since a copy of the object occupies the same space as the original, it is a good idea to move it a bit to make sure that there are actually two objects.

Example:

- Load a half cylinder from the Primitives menu. Scale it up by all three axes. Then, turn X and Y off and flatten the half cylinder by scaling down the Z axis.

- Select Copy from the Tools menu. The half cylinder will blink. Now select Rot Y (ObjAx on) and move the mouse. The second half cylinder will rotate smoothly around its edge.

Purpose

To create a mirror copy of an object.

Procedure

1. Select an object to be mirrored

2. Select the tool menu

3. Select "Mirror". The object will be mirrored along world XZ plane.

Note:

Often the mirror is used to created symmetrical objects. The user would design on half on an symmetrical object (half a face) copy it , mirror the copy and glue mirrored copy to the original to obtain a full face.

Since there is only one topologically different object that can be created with the mirror operation, it is not necessary to specify the plane around which the object will be mirrored.

Negate

Purpose

To reverse surface normals of all polygons of an object.

Procedure

1. Select an object

2. Select the tool menu

3. Select "Negate".

Note:

Negate is used mostly to correct objects which were converted from Sculpt or Lightwave formats and perhaps decomposed. On some subobjects the surface normals will be facing inside instead of the outside of the object. Negate will correct this and ensure that an object will be rendered properly in quick or broadcast renderer.

Decompose

Purpose

To separate an object with disjunct sets of faces into separate sub-objects.

Procedure

1. Select an object to be decomposed

2. Select the tool menu

3. Select "Decompose". The object will blink once to show that it has been decomposed.

4. Select "\/" to see the newly created hierarchy.

Note:

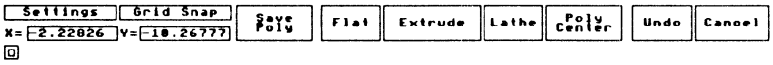
There are two cases when the user may wish to decompose an object. Firstly, some foreign objects (such as Videoscape Lotus) even though discontinuous, may not have hierarchical structure and decompose operation will create one. Secondly, in Caligari, a slice separate operation may create disconnected objects which also do not posses hierarchy.

Decompose is executed automatically after reading a Sculpt object.

2D(2)

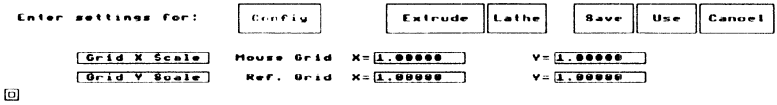
EXTRUDER

The "Extruder" enables the user to draw any 2D shape by hand and sweep it through 3D space to create a 3D object.



Extruder can be entered from the Tools menu. Upon entering the Extruder the screen presents a 2D grid. The user can draw the shape simply by rubberbanding a line segment and by depressing the left button to create a new point.

If Grid Snap is selected, new points will be snapped at the intersections of the background grid. The size of this grid can be set with "Settings".



The bottom of the menu continuously displays the X and Y coordinates of the cursor. A new point can also be entered numerically by selecting the X or Y field and by typing the coordinates from the keyboard. This is useful for entering shapes from blueprints with precise, known coordinates.

Undo will remove the last active line segment from the drawn curve, so that an error can be easily corrected. When the shape is finished, the "**Close Poly**" button must be selected. This automatically closes the polygon by connecting the first and the last point. **Caligari** will not allow for creation of self-intersecting polygons and the program checks for these continuously as the polygons are drawn.

After closing the polygon, **Extrude, Flat and Lathe** options appear on the menu. The user can select one of them or save the created polygonal shape, draw a hole inside the polygon, set the polygon center or edit the polygon using the Undo option.

Fonts can be quickly generated in the Extruder and saved as polygonal shapes rather than 3D objects.

The three options for 3D sweep can be modified by user selectable settings such as **Orientation, Depth, Shear, Scale, Auto Sub, Reference** and **Mouse Grid, # Segments** and **Angle** (see Settings).

The **Extrude** option can assign a specified thickness to a polygon. The **Flat** option will generate flat objects (without thickness). **Lathe** will sweep the polygon along a circular path.

Purpose:

To adjust the extruder grids.

Procedure:

1. Select "Extruder" from the Tools menu.

2. Select "Settings" from the Extruder menu.

3. Set the Mouse or Reference grid size from the keyboard or grid X or Y Scales.

4. Select "Use" or "Save". The program will return to the Extruder menu with the new configuration settings.

Note:

Grid X Scale and Grid Y Scale - the Extruder will display horizontal and vertical rules for drawing.

Reference grid - provides visual help by displaying the grid.

Mouse grid - actually specifies where the polygon vertices can be placed. Mouse and reference grid need not have the same values.

Example:

- Set the mouse grid to X = 0.5 and Y = 0.5 and select "Use". Now the mouse step will be half that of the Reference grid.

Extrude Settings

Purpose:

To adjust the extrude settings.

Procedure:

1. Select Settings from the Extruder menu.

2. Select "Extrude" from the Settings menu.

3. Set Orientation, Auto Sub, Depth, "Cone" Scale, Shear X or Shear Y.

4. Select Use or Save. The program will return to the Extruder menu with the new Extrude settings.

Note:

Orientation - will position the extruded object in the world aligning it with the X, Y or Z world axes.

Auto Sub - when selected, will display the settings automatically each time the extrude operation is selected.

Depth - will specify the thickness of the extruded shape.

"Cone" Scale - scales the polygon being extruded. If the user draws a circle, the default value of 1 creates a cylinder, values less than 1 create a cut cone and a value of 0 creates a cone.

Shear X and Y - shift the top polygon horizontally or vertically. If the value is non-zero the object will appear to "lean" to one side.

Extrude

Purpose:

To give a thickness to a designed shape.

Procedure:

1. Select "Extruder" from the Tool menu.

2. Draw or load a 2D shape.

3. Click the "Close Poly" button. If desired, draw any number of holes inside the shape.

4. Optionally change Orientation, Depth, "Cone" Scale, or Shear X or Y (from Settings menu).

5. Select Extrude. The program will return to the Navigation menu with the newly created 3D object highlighted.

Note:

Selecting the "Close Poly" button will return a "Polygon invalid" message if the first and last points cannot be connected without intersecting some other polygon edge. Use Undo to modify the polygon until it can be closed.

The total number of edges of a polygon and it's holes cannot exceed 300. A polygon cannot have more than 29 holes.

For rendering purposes Caligari will assign a different material to the front and back of the extruded shape so that in attribute menu the user can for example make front side of extruded letter complex facet appearance and sides smooth appearance.

Example:

- Draw a circular shape. Draw two holes inside it. Notice that a hole cannot cross the boundary of its surrounding polygon. Select Extrude.

Purpose:

To create a flat object.

Procedure:

1. Select Extruder from the Tool menu.

2. Draw or load a 2D shape.

3. Select the "Close Poly" button. If desired, draw any number of holes inside the shape.

4. Select "Flat". The program will return to the Navigation menu with the newly created flat object highlighted.

Note:

Flat objects are useful for adding detail on the surface of other objects, to create roads, fences, etc.

Lathe Settings

Purpose:

To adjust settings for Lathe.

Procedure:

1. Select Settings from the Extruder menu.

2. Select "Lathe" from the Settings menu.

3. Set "Orientation", "# Segments", "Total Angle" or "Auto Sub".

4. Select "Use" or "Save". The program will return to the Extruder menu with the new Lathe settings.

Note:

Orientation - will position the lathed object in the world aligned with the X, Y or Z world axes.

Auto Sub - when selected, will display the settings automatically each time the Lathe operation is selected.

 # Segments - designates the number of segments into which a circular path is divided. If it is a large number, the object will appear to have a smoother surface, but the rendering time will be longer. This will particularly effect the first quick rendering of the object due to the algorithm **Caligari** uses for depicting solids.

Total Angle - If 360 (default is selected), the polygon will be swept a full circle. If less than 360, the object will be swept only part of the circular path. If the initial polygon had holes in it, these will be visible at the ends.

Purpose:

To sweep the polygon design through space around any edge or specified axis.

Procedure:

1. Select Extruder from the Tool menu.

2. Draw or load a 2D shape.

3. Select "Lathe" from the Extruder menu.

4. Select an edge of the polygon or draw a line to be used as the rotational axis.

5. Specify #Segments and Angle for rotation or use the default values.

6. Select Go. The program will return to the Navigation menu with the newly created 3D object highlighted.

Note:

If 360 degrees for the rotational angle is specified, holes inside the polygon will be eliminated. If a line is drawn for the rotational axis, **it must not intersect the polygon**.

For rendering purposes Caligari will assign a different material to the front and back of the lathed shape if less than 360 degrees is specified. Thus in attribute menu the user can assign to the both ends of cut shape faceted appearance while the sides will have smooth appearance.

Due to the algorithm used in Quick Render, objects lathed with a lot of segments (more than 32) may cause a crash during the rendering process. They will however, render fine in broadcast render.

Example:

- Draw a small circular shape. Select Lathe and draw a line outside of the shape. Select Go. A donut shape will appear in the workspace.

Load/Save Poly

Purpose:

To load and save a 2D shape.

Procedure:

To load:

1. Select Extruder from the Tool menu.

2. Select "Load Poly" from the Extruder menu.

3. From the menu of available shapes select one and press "Load".

4. The selected shape will appear in the extruder for further manipulation.

To save:

1. Draw a 2D shape.

2. Select "Save Poly". Type a name for the current shape and press "Save".

Note:

Fonts should be saved as 2D polygons. A 3D word can be quickly assembled by loading the letters into the extruder and specifying the desired thickness. Also, maps can easily be saved in this form since a 2D form takes up very little space.

Example:

- Draw a small circular shape and save it as a polygon. Extrude the shape into a cylinder. Return to extruder and load the same shape again. Now use Lathe (set the angle to 180 degrees). Align both 3D objects in the workspace along the identical outline and glue them together. With the proper scaling, half a donut and a cylinder glued together produce a nice umbrella handle.

Poly Center

Purpose:

To set the center of an object's coordinate system

Procedure:

1. Select Extruder from the Tool menu.

2. Draw or load a 2D shape.

3. Select "Poly Center".

4. Position the large crosshair cursor and depress the left button.

Note:

If you rotate an object, it will rotate around the center of its coordinate system.

Example:

- Draw a large polygon with a sharp end. Select "Poly Center" and set the large crosshair on the sharp tip of the polygon. Extrude the polygon. Now rotate it in worldspace around its Z axis. The extruded polyhedra will rotate around the point you just set.

2E

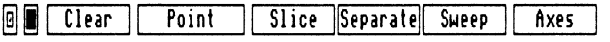
2E. SINGLE POINT EDITING

Single Point Editing allows the user to select a vertex, an edge, a face or a part of an object and then manipulate them with the navigation functions move, rotate, and scale as well as with a few other operations (sweep).

The user can also create a new vertex on an existing edge, a new edge or edges as a crossection of a face, and a new polygon or polygons on a crossection of an object (vertices can't be deleted).

The user can also create and manipulate multiple points, edges, faces or parts of objects. Example of this would be the sweep of several faces simultaneously, with subsequent scaling or rotation of all swept surfaces.

The single point editing is accessible through the "PEdit" button on the base menu. Selecting "PEdit" button in the base menu will bring up a new "PEdit" menu with all of the controls for single point editing.



Selecting the "Point" button in the edit menu puts the mouse in the "Point" mode. Moving the cursor into the design space and pressing the left mouse button will select a point on the object under the cursor.

By selecting single points, a special object called the "edge set" is created. This object is highlighted just like any subobject of currently selected object. Its shape depends on the number of selected points.

One Point- If one point is selected, edge set consists of one vertex. It may be an existing vertex or a new vertex created on the edge under the cursor.

Two Points - If two points are selected, edge set consists of one or more edges lying on the line between these two points. Both points must lie on the same face, otherwise the second point is not accepted. An existing edge can be selected this way by selecting its two vertices. If points are not already connected by an edge, a new edge or edges are created on the crossection of the face and the line between selected points.

Three Points -If three points are selected, the edge set consists of a face or a crossection of an object. If all three points lie on the same face, this face creates an edge set.

In this case, the third point must be selected on a vertex, not on an edge between vertices. If this face has some holes and all three points lie on the same polygon, none of the holes will be included in this edge set. This allows for manipulation of a single hole or a face without its holes.

If the third point doesn't lie on the same face as the first two vertices, a crossection of the whole object by the plane defined by these three points will create an edge set.

This crossection must consist of closed polygons, otherwise the third point will not be accepted.

Four Points - If four points are selected, the edge set consists of all the parts of an object lying in the same half space along a cross-section plane as the fourth point.

Fourth point is accepted only if first three points create a crossection of an object and fourth point is above a vertex lying outside of the cutting plane.

Last selected point can be deselected by clicking on the "Undo" gadget. Any new vertices and edges, created by last point selection, are also erased. Only the last selected point can be undone.

If an edge set is created, the "PEdit" gadget in the base menu is selected. This is true even if "PEdit" menu is closed, or an edge set is not currently active.

The last case may happen, when edited object is activated by selecting the "Up" gadget in order to manipulate to position an edge set better. To reselect the edge set, just select the "Point" gadget again.

Once selected, an edge set can be moved, rotated, and scaled just like any other object. An edge set has its own axes and a center, which can be distinct from those of an edited object. The center of an edge set is calculated when next point is selected as the center of all vertices in the edge set. If one or two points are selected, "Z" axis orientation is calculated as an average of surface normal of all adjacent faces. When third point is selected, "Z" axis is calculated as the normal of a plane defined by three selected points, i.e. "X" and "Y" axes lie on this plane.

The axes of an edge set can be displayed and manipulated by selecting the "Axes" button. Once selected they can be moved and manipulated just like any other object.

If an edge set is active and "ObjAx" is selected, the navigation operations will operate along an axis described above. But if "World" is selected, they will not operate in world space, but in object space of edited object. Additionally the rotation of an edge set is executed around the center of the edited object.

Face polygons in Caligari objects are supposed to be planar. This is necessary for good rendering and also for many calculation in point editing. Therefore if, as a result of a navigation operation, any vertex from edge set is moved out of the plane of the face it belongs

to, this face will be triangulated. If "Undo" of this operation is selected, the triangulated edges are deleted too.

You can switch to subdividing to quadrangles instead of triangles by clicking on "Triangles" button in NUM menu. This subdivision is more time expensive, it creates much more new faces, but it often creates very nice smooth subdivision.

It is also possible to turn off the subdivision by clicking on "Quadrangles" button in NUM menu. In this case the full responsibility of keeping the polygons planar will lie on the user. If the user creates a non-planar face, it maybe not be possible to slice it or several other operations as well as achieve correct rendering.

Note:

Dynamic pick facility doesn't operate with navigation operations while any part of an object is selected for editing.

ERASE

If the user selects the "Erase" button, any selected part of edited object is deselected. This is true, even if edge set is not currently active (e.g. if edited object is activated by selecting the "Up" button). The "PEdit" button in the base menu will be unhighlighted.

Any new vertices or edges, which were created by points selection, will remain in the edited object. They can be erased only by selecting the "Undo" immediately after their creation by points selection or another editing operation.

SLICE

A crossection of an object can be created by explicit definition of a plane by selecting three points in "Point" mode as well as a crossection of a face by selecting two points on it in "Point" mode.

The Slice function is useful to create a series of crossection of an object parallel with each other or with one of the faces of the object.

This is useful for moving or scaling of these crossection, because transformations of parallel slices do not destroy the planarity of adjoining faces.

First a face or a crossection of an object must be selected to define a cutting plane, or an edge or a crossection of a face must be selected to define a cutting line.

Selecting the "Slice" button puts a mouse in "Slice" mode: Placing a cursor in the design space and moving a mouse with depressed button, will move the cutting plane across the object or a cutting line across the face. When a button is released, a new crossection of an object or a face is created.

When a cutting plane leaves an object or a cutting line leaves a face, it is stopped in the last visible position. First reverse move of the mouse causes actual move of a cutting plane, regardless of how much the mouse has been moved after the plane was stopped.

Immediately after the slicing is done, a crossection can be moved, rotated or scaled or another parallel crossection can be created. The crossection can also be erased by selecting "Undo".

SWEEP

If an edge set is selected, it can be extruded from the edited object. The edge, the face or the crossection of an object are copied, whole edge set is moved away from the object, and corresponding vertices are connected by new edges. The move away from the object is executed along "Z" axis of the edge set, and if the axes haven't been manipulated yet, it is perpendicular to the face or the crossection of an object. If the edge set contains insulated vertex, it is only moved along "Z" axis without copying it.

If "Undo" is selected immediately after "Sweep", extruded face or part of an object will move back to the original plane but the sweeping operation will be preserved for later scaling of the edge set and perhaps selecting another "Sweep" on the scaled face.

SEPARATE

Any selected edge set except for a vertex can be separated from the edited object by selecting the "Separate" button. This operation thus creates one new object (if less then four point are selected, only the outline will be separated).

Notice, that since Caligari objects can be concave and have holes in them, separation can result in non continuous objects. These can be decomposed to form a hierarchical object (see "decompose").

AXES

Selecting the "Axes" button will display object's axes. Axes can be displayed for any single or complex object or any subobject as well as for an edge set selected with single point editing. Each complex object has its own axes, distinct from the axes of any of its subobjects.

The axes can be moved to redefine the center of an object or they can be rotated to change the orientation of the axes.

Selecting "Axes" button again will make the axes invisible. If instead the "Up arrow" button is used, the object can be manipulated with displayed axes.

Any number of axes in a scene can be displayed by using the Dynamic Pick and "Up" and "Down" gadgets, and multiple use of the "Axes" button. To reselect the axes as active objects, to manipulate them or to make them invisible, the object with displayed axes must be selected, and then the "Axes" button must be selected. To remove the axes, they must be reselected as above and select "Axes" button again.

QUADDIVIDE

Clicking on the **QuadDiv** button will subdivide selected faces in the edgeset. If no edgeset is selected, all the faces of the selected object will be subdivided. Repeated clicking on the QuadDiv button will produce subsequently finer subdivision of selected faces.

MULTIPLE POINT EDIT

The user can select multiple points, edges, faces or parts of objects as well as multiple slices. To create a new edge-set, the user must click the **right mouse button** on the desired point.

Clicking the left mouse button selects new elements within the current edge-set (as before). The user can also select a "poly-line" by selecting a point and then clicking on a connected edge. An arbitrary face set bounded by a poly-line can be made by first closing the poly-line and then clicking on any point or edge inside the poly-line.

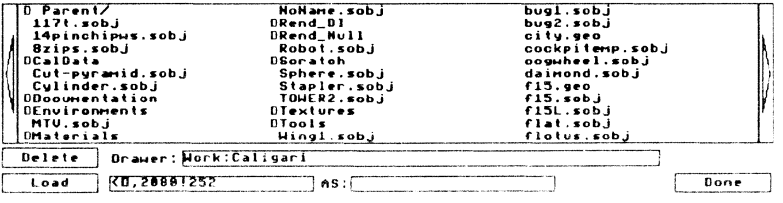
Manipulation of multiple edge-sets in the object space will cause to move vertices, edges and unclosed poly-lines along the average surface normal of faces which contain them. Face set will move perpendicularly in a manner similar to that of slice.

The user can separate or sweep anything except a vertex. To select multiple parallel slices, slice with the right button rather than the left. The current slice will be copied and the previous one will stay in place.

2F

Load

The "Load" menu is accessible from the Base menu. When selected, it replaces the current menus but leaves the workspace unchanged.



The purpose of the Load menu is to display the objects available for loading into the workspace.

The Load menu can display the names of up to 44 objects and directories simultaneously in four columns of eleven rows each. If the current directory contains more objects, the user can scroll the menu horizontally, using the arrows on the sides of the displayed object names.

At present Caligari can load objects in its own format as well as in three foreign formats: Videoscape, Lightwave and Sculpt3D. Compatibility with these most popular Amiga formats is an important feature for all the users utilizing other Amiga 3D packages in conjunction with Caligari.

Videoscape (text 3DG1 format) - In order for a Videoscape object to appear in the file menu it must have the extension "GEO".

Lightwave - Extension LWOB is required

Sculpt - Extension SCENE is required

More formats will be supported in the future.

Selection of objects from the menu is simple. Place the cursor in the area where the name of the object you wish to select is displayed and depress the left mouse button. The object under the cursor will be highlighted. Now move the cursor around and, as it moves, the object names under the cursor will be highlighted. Release the but-

ton and the newly selected object will appear in the bottom line next to the Load button.

Selecting Load projects the highlighted object into the work space. The object appears at world coordinates (0, 0, 0), and is **normalized,** i.e., its local coordinate axes are aligned with the world axes.

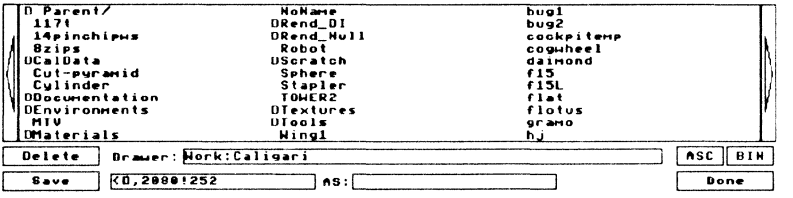
The user can also load the selected object under a different name by selecting the "AS" field (next to "Load" field) and typing in the name under which the object will appear in the workspace. For example, you can load the object "ROBOT" as "ROBBY" the first time and as "ROBBIE" the second time so you will be able to distinguish one robot from the other.

If an object is loaded into the workspace several times without using the AS option, **Caligari** automatically assigns a different name to each new introduction of an object. This is done by placing an exclamation point after the name, followed by a number; for example, ROBOT!2.

If a new object is created (in the extruder module or from primitives), **Caligari** assigns it a default name, optionally followed by a number. The default name is "NoName". The user can change this default name when the object is saved.

Save

The "Save" menu displays the names of all the objects in the current directory.



The Save menu is similar to the Load menu. The "AS" option allows renaming of objects prior to saving them. If the "AS" option is not specified, the object will be saved under its current or default name.

Currently Caligari will save objects in **Videoscape** or **Lighwave** format in addition to its own format. To select format other then Caligari format, the rolling button must be activated.

If an object with the specified name already exists in the current directory, and the object in the current directory has not been loaded since entering object design, **Caligari** displays a new menu asking if you want to overwrite the existing object in the directory with the new one.

Example:

- Load the cone from the Primitives menu and scale it up to make it larger. Select "Save" while the modified cone is still highlighted. The Save menu will be displayed.

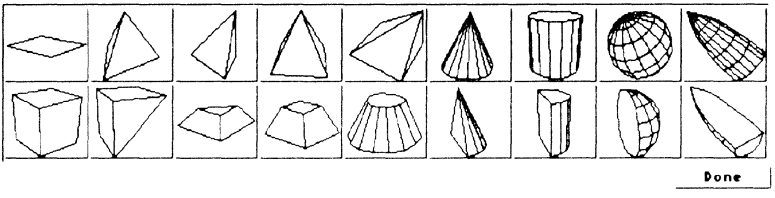
- The word "Cone" will appear next to the "Save" button. Select the field next to "AS:" and type in "NewCone". Select the "Save" button and the modified cone will be added as "NewCone" to the current scene.

- Exit the Save menu and enter the Load menu. The newly saved object will be displayed here together with other objects ready to load into Object Design or Scene Composition.

2G

Primitives

"Primitives" are a group of basic 3D volumes - cube, cone, sphere, etc - available for instant loading into the workspace. Because the Primitives are used so often, their actual shapes are shown rather than their names.



The Primitives menu is similar in many ways to the Load menu, but some differences exist. Primitives are displayed on the menu in two rows of nine elements each. When selected, a Primitive appears in the workspace in a normalized position (0, 0, 0) in the worldspace and with its local axes aligned with the world axes.

Even though the number of primitives is small, they are very powerful building blocks. For example, a Cube can be stretched to represent a table top or squeezed to represent one leg of a table. Using the Single Point Edit and combining simple forms by "Gluing" can produce an enormous variety of complex shapes.

2H

Render

Wireframe images of the 3D scene are useful representations of the actual design, but there are times when a solid and shaded image is necessary. At any time in Object Design, a solid rendering of the currently active object can be obtained by selecting "Render" (this is Quick Render) from the Base menu.

When an object is rendered for the first time, **Caligari** needs to do extra computations to create the data necessary for solid representation resulting in a few seconds delay. After that, however, the user can switch between the solid rendering and wireframe representation almost instantaneously.

In Render mode all menus are eliminated to provide unimpaired viewing of the object. You can exit this view by **clicking the right mouse button.** This will bring up four buttons on the bottom of the Render mode: **Print Image, Save Image, Color** and **Exit Render.** Using these functions, the user can print the image on the printer, save it as an IFF file, change the color of the object or return to the wireframe mode.

The solid view in Object Design is primarily intended to give the user a clearer understanding of the geometry of the object as well as facilitating assignment of its colors. As such, it has only one fixed light source and provides for a diffuse shading. Unlike the scene module, the user can shade only one object (selected) at a time. On the other hand the Scene module provides for multiple lightsources and diffuse as well as specular shading of all objects in Quick or Broadcast renderer.

Last important use of the solid view is color assignment to objects, their subobjects and individual faces.

Example:

- Create a simple object in the extruder. Alter the viewpoint for a good viewing angle and select Render. After a few seconds a solid image of the object will appear.

- Double click the right button. This will bring the menu up. Select "Exit Render". Change a different view in the wireframe and select "Render" again. Notice that this time the object will be rendered almost instantaneously.

Print Image

Purpose:

To obtain a printed copy of the current image.

Procedure:

1. Select Render from the Navigation menu.

2. Click the right mouse button after the image is rendered.

3. Select Print Image and your image will be printed on the printer currently selected in Workbench Preferences.

Note:

Make sure that you set the correct printer driver from Workbench Preferences and that your printer is connected to the computer and turned on.

Save Image

Purpose:

To save an image in IFF format.

Procedure:

1. Select "Render" from the Navigation menu.

2. Click the right mouse button after the image appears.

3. Select "Save Image". A Save submenu will be displayed.

4. Select "Name" and "Drawer" for identifying and storing the image. Select the "Save as IFF File" button.

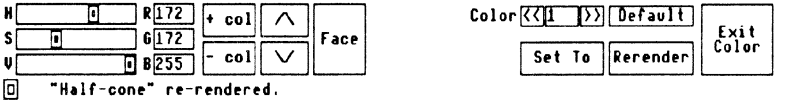
Note:

Caligari can save images in medium or high resolution. An IFF image can be manipulated in a variety of programs such as Deluxe Paint, Photon Paint, etc.

2H(1)

Color

The Color menu enables the user to interactively assign a color attribute to an object, or to an individual faces of object, as well as to change its visual appearance on the screen.



Color attributes are represented by several horizontal color bands (palettes) just above the color menu (the default number is one plus background and the color of the light).

A color attribute can be assigned to an object, subobject or individual face by selecting the "Set To" button from the color menu. The currently active color palette is first assigned to the selected object. **Clicking** the "Set To" button will immediately rerender the selected object or subobject in the assigned color.

Since the number of available palettes is limited, the user can see the correct color of the selected part of the complex (hierarchical) object using "**Exclusive**" option. "**Shared**" option will attempt to select the best palette combination for all the object colors. If there are too many attributes assigned to the object (such as when two multicolored objects are glued together), the results may be poor.

Just as in wireframe, the user can choose any subobject by selecting up or down (/\, \/) arrows. Once the subobject is selected, the rest of the object is shaded out, highlighting only the subobject the user wishes to color.

The visual appearance of the object on the screen can be modified by varying the color palettes. The leftmost palette represents the color of the background, second from the left controls the color of the highlights (color attribute 0) and should not be used in object design. The third, fourth and consecutive bands represent in shaded gradation the appearance of first, second, and consecutive color attributes.

The color range of each palette can be changed with the three horizontal sliders on the left side of the color menu. The top slider

changes the color (hue), the middle slider changes color purity (saturation) and the bottom slider changes the intensity (Value) of the palette. The number of palettes (color attributes) can be changed using "+ col" and "- col" buttons.

A palette can be activated by positioning the cursor on top of the palette and clicking the left mouse button. Clicking the right mouse button on top of the palette and moving the mouse will change the number of shades in the palette. If the user increases the number of shades on any palette (for smoother appearing objects colored in one attribute), the number of shades in the other palettes is automatically reduced. Also, if too many palettes are added, each of them will have only a few shades. Thus, the user can trade number of color attributes (palettes) for smoothness of shading.

Caligari will try to preserve the appearance of each attribute for Broadcast render. The appearance of some attributes may be different in Object Design and Scene Composition (and in different scenes). Color attribute **number**, however, is permanently assigned. You can think of it as a "material" the object is made of. For example, if in object design the car wheels have been assigned attribute one and the car body attribute two, these materials never change. However if we have two copies of the same car in the scene design, then one body can be red and other green (material appearance). Since all the wheels are assigned the same material (one), one wheel of a car could not have a different color from another wheel on the same car.

A color attribute can be assigned to a whole object or to any subobject or to any subset of faces of any subobject. The user can choose any subobject by selecting "Up" or "Down" gadgets with mouse dynamic pick facility, just as in wireframe.

Once the subobject is selected, the rest of the object is shaded out, highlighting only the subobject the user wishes to color.

FACE

To assign color attributes to an individual faces, the "Face" gadget must be selected. Face mode of color menu will appear. Instead of "Exit Color" button, there is "Exit face" button in this menu.

To select a face in face mode, point the cursor over the face you want to select and depress the left mouse button. To deselect an already selected face, depress the button once more. Any number of faces can be selected before assigning a color attribute to them.

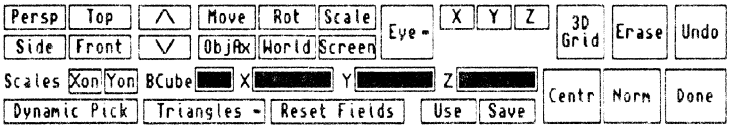
To assign a color attribute to selected face or faces, select the "Set to" button. After selecting "Set to" whole object is rerendered, i.e. selected faces are colored the newly assigned color. No face is selected after setting attributes.

Faces can be selected only from a simple object. If current object is complex, first simple descendant of this object will be highlighted and the rest of the object will be shaded out.

The user can reverse the normals of selected faces using "Reverse" option. This is generally useful for correcting bad surface normal of Sculpt objects.

21

"NUM" is a submenu of the Navigation menu which enables the user to enter data numerically, and other utility functions. This menu is identical in Object Design and Scene Composition. When selected, it replaces the current menu configuration with its own menu display.



Settings - includes threshold for the bounding cube and rulers for orthogonal views.

EYE/OBJ - allows a numerical entry for the viewpoint position or position of selected object.

3D GRID - allows the user to move objects in predefined regular steps and set the value for this step.

Various utility functions - **Normalize, Center, and Undo.**

Caligari tries to de-emphasize numerical entry of data. It is very important, particularly in the conceptual stage of design, to maintain real time feedback between the hand and the eye, which is impossible when using keyboard entry. However, there are a number of cases, where the ability to enter precise numerical data (perhaps from an existing blueprint) will enhance the functionality of the software.

Cube

This is a threshold for switching between the full representation of the object and its bounding cube. It is set in sixtieths of a second, so the default setting of 12 sixtieths of a second is equivalent to five frames per second. If an object cannot be updated five times per second only its bounding cube will be shown. This forms the basis of the real time response of **Caligari.**

The user can set this threshold to lower values (object will be replaced with a bounding cube sooner, resulting in a smoother interaction) or higher values (object will change into a bounding cube later, resulting in a jerkier motion of the full object). This can be particularly useful if the user has a faster processor in his computer.

Scales

These are dynamic scales which are displayed horizontally (X) or vertically (Y) on the screen. Scales help the user to orient himself in the design space and to position objects with a higher degree of precision. As the user manipulates the design environment, the scales update themselves.

Both scales are available in top view, X (Y) scale is available in side (front) view and no scales are available in the perspective view.

Subdivision

Some editing operation such as Single Point Edit may cause a loss of planarity on some polygons. This is generally undesirable as it may result in rendering errors. Caligari will therefore break offending polygons into **Triangles** or **Quadrangles** if the user selects corresponding option. The user can also choose to turn the automatic subdivision off.

Transformation operations are numerically performed in the same way as in Navigation menu, only the values are obtained not from a mouse move but from numerical entry for x, y, and z.

Procedure:

1.Select the Num gadget from Navigation menu.

2.Select the desired transformation function Move, Rot, or Scale and select Eye/Obj.

3.Select desired coordinate system ObjAx, World, or Screen and select desired axes X,Y,Z.

4.Enter X,Y, and Z values from keyboard or use the default values.

5.Move the cursor to the design space and click on left button.

Note:

To perform desired transformation, move the cursor to design space and click on the left button. If an object transformation or an eye transformation in the object coordinate system is specified, first the object under the cursor is selected and then the transformation is performed. This allows the user to perform the same transformation with any number of objects in a scene simply by clicking on the desired objects, as well as to perform the same transformation several times sequentially with the same object by multiple clicking on this object.

By default, numerical values are relative to the current position of the eye or the object. If absolute move or absolute rotation of the object is required, first select Centr to move the object to the origin of current coordinate system, or select Norm to align active object with the axes of selected coordinate system.

Predefined values for numerical transformations can be changed by selecting "Use" or "Save" buttons after entering required values in "X","Y","Z" numerical fields. ("3D grid" must be deselected).

This button will position the object at the center or it will turn the view to the center of the specified coordinate system.

If Eye is selected, selecting Centr will turn the view to the active object or to the World center (0,0,0) according of ObjAx or World selection. No action will be performed while Screen coordinate system is set, because the view already sees the center of the screen. This operation can be used to find a lost (selected) object.

If OBJ is selected, clicking on Centr will move the active object to the World center (0,0,0) or to the center of the screen based on the current coordinate system. No action will be performed in the object coordinate system, with an exception of special objects like "Axes" or "Edge set"(see Single Point Editing). Axes are centered to the center of bounded cube of the object. Edge set is centered to the center of the currently edited object.

If Centr is used immediately before an object is moved, an absolute move of this object is obtained.

This button will align the object with the currently active coordinate system, or it will reset horizon and scale factor of the view.

If the Eye is selected with the Object or World coordinate system, selecting Norm will align the eye according to "X","Y","Z" buttons selection. If only "Z" is selected, the eye will be aligned horizontally. If "Y" is selected the eye will be align with world vertical. If "X" is selected the eye will look along world X axis. For practical usage select Z only to level the eye or select all three axes before absolute encircle (Rot with ObjAx or World) is required. If the Screen is selected, Norm will scale the eye to the default scale factor.

If the OBJ is selected, selecting Norm will align the active object with the World axes or with the screen axes according to the current coordinate system. No action will be performed in the object coordinate system, with exception of special objects, like "Edge set" (see Single Point Editing). Edge set will be aligned with the axes of edited object.

If Norm is used immediately before rotating an object, absolute rotation of this object is obtained.

If the "3D Grid" button in Nav menu is selected, the transformations (move, rotate and scale) are performed in predefined regular steps. Num menu allows the user to change these predefined values. This can be done by selecting the X,Y,Z numerical fields and entering values from the keyboard. Each object and eye transformation has its own grid value and they all can be changed separately.

Procedure:

1. Select the Num button from the Navigation menu.

2. Select the 3D Grid button from Num menu.

3. Select desired transformation function Move, Rot, or Scale and EYE/OBJ.

4. Enter X,Y, and Z values from keyboard or use the default values.

5. Select Use or Save to change grid values momentarily or for all future uses.

Note:

The grid can be turned on selectively with free motion in one or more axes and step motion in the other axes. To specify which axes are stepped, select X,Y,Z gadgets while 3D Grid is selected. This set of stepped axes is stored separately for each object and eye transformation.

If the grid for eye scale is changed, only the X numerical entry will be accepted.

Only positive values can be used or saved. Negative values are automatically changed to positive.

3

SCENE COMPOSITION

The Scene Composition module enables the user to build a complex scene out of the objects created (and saved) in the Object Design module. With it, one can also assign surface properties to the objects and light the scene with multiple light sources.

Ideally using Object Design, the designer will build the objects he or she thinks will be useful for the final scene. When enough objects are built, the user will save them and then enter the Scene Composition module to assemble the scene. Here, he can position the objects, scale or rotate them and also duplicate them to compose a larger scene such as a "living room" or a "space station".

Since it can have a great impact on the final result a great deal of attention should be spent on setting up the lighting of the scene. Unlike the paint programs, **Caligari** does not deal with pixels directly; rather the user "paints" with light. Skillful use of color attributes and lights can produce striking results.

With **Caligari,** one enters the Scene Composition module with the Base menu and the Navigation menu present. One sees the workspace viewed in perspective with the Rot/Eye function active and the X,Y,Z coordinate axes selected, much like in Object Design. A selected function will be indicated by an amber highlight.

In the Scene Composition module the objects can not be modified, but they may be moved, rotated and scaled. Hierarchical access is inoperable in this module. If the user wishes to edit an object, he must return to the object module to do so.

In order to fully utilize the Scene Composition module it is important to realize that, unlike in Object Design, when an object is loaded, it does not explicitly load its geometrical data. Instead, it only sets a pointer to it. Additionally, each scene object has data for its appearance attributes and a matrix for its position in the world, its rotation and scale.

Thus, if several copies of the same car are loaded into the scene, each can have its own color, placement (translation), rotation and scale, but they all share the same geometrical data (a set of vertices and a hierarchy of polyhedra). This has two advantages - memory is preserved, so that a very complex scene (using many copies of the same object with different scale and rotational factors) can be built, and second, time is saved because the compilation process for the first solid view rendering (in Quick Renderer) is performed only once for all copies of the object.

On the other hand, if several copies of the same object are loaded into the scene and the original object is modified in object design, then, upon entering the scene, all the copies will be modified too. If the user places two robots in the scene and one has its arm pointing up while the other has its arm pointing down, two different robots must be created in Object Design and then named differently.

In brief: it will often be possible to use the same object many times in a scene e.g. a building may be colored, scaled and rotated to appear as several different buildings. Also, it makes sense to use smaller objects rather then large ones, since the compilation time for solid rendering is much shorter for smaller objects.

3A

BASE MENU

The Base menu is always present on the bottom of the screen. It can be made invisible (along with other menus) by selecting its close button. It can be made visible again by clicking the right mouse button.

The arrangement of buttons in the Base menu is slightly different from that in Object Design.

These are the major functional choices in **Caligari**. When selected, some will activate additional menu strips, some will replace existing menus, and some will open completely new screens.

NAV - Navigation is a central part of **Caligari**'s interface. It enables the user both to move his/her viewpoint and manipulate objects in the 3D environment.

ANIM - Displays the animation submenu

TOOLS - Provides access to the STATUS, LIGHTS and COPY functions.

ADDOBJ - Displays the menu for loading objects into the current scene.

FILE - Permits loading, saving and deletion of scenes.

RENDER - Displays the current scene in a solid view (in color) using quickrender.

BRENDER - Enters the Broadcast quality renderer menu.

OBJDES - Allows the user to exit the Scene Composition module and enter the Object Design module.

QUIT - Allows the user to exit the program.

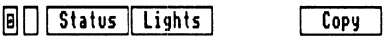
Both the Navigation and the Status menus are identical to those of Object Design. The Tools menu, however, is different.

3B

TOOLS

The Tools menu works in conjunction with the Navigation menu. It is simply added to the stack of existing menus.

Three "tools" become available: **Status, Lights** and **Copy.**



Status - will display the information about the current object and the viewpoint. It is identical to the object design status menu.

Lights - allows for creation of lighting effects within the scene. With this tool, the user can place any number of lights in the scene and assign their intensity and softness.

Copy. - permits the user to create copies of objects.

Copy

Purpose:

To create one or more copies (clones) of an object.

Procedure:

1. Select an object.

2. Select "Copy" from the Tools menu.

3. The newly created object will occupy the same space as the original object.

Note:

The new copy of an object will be superimposed over the original; Therefore, it is good idea to move it a bit to verify that there are actually two objects.

The Copy operation is different from that in Object Design. It does not create separate data for each new object, rather, it points to the original object stored in the database. A clone does not have its own set of geometrical data (vertices, faces) but it can differ in its color, diffuse and specular coefficient as well as size.

This saves memory, particularly when large scenes with many objects are created.

3B(1)

Lights

The purpose of this menu is to enable the user light the scene for solid view in the quick renderer. **Caligari** allows for an unlimited number of light sources. However, since it produces flat (faceted) shading (in Quick Render), the number of lights should be minimized since too many lights would simply cancel each other.

Another thing to remember is that even the faintest and most ambient light will produce a specular highlight on a shiny object. Too many highlights will not look realistic on a flat shaded object. On the other hand, skillful use of lighting together with specular effects can improve the results substantially. More elaborate lighting is available in the Broadcast renderer.

The Lights menu uses a numerical form of entry. The following items are on the menu:

Light #- Indicates light whose parameters are displayed.

> > - Allows the user to scroll among the defined lights.

X, Y, Z- This is a set of X, Y, and Z coordinates of current light with full floating point precision.

Intensity - In this function only relative intensities are meaningful since **Caligari** automatically adjust specified intensities into the available Amiga range of color space. Thus, if there are two lights, relative intensities (1, 10), (0.1, 1) would produce the same lighting of the scene.

SoftSize - specifies softness of the current light. Note that Intensity and softsize are ignored in the Broadcast renderer.

0 - point light source, all polygons not facing the light are completely dark.

1- planar light source, only polygons facing exactly the opposite direction from the light are completely dark.

1000 - ambient light, polygons in any direction are illuminated with the same intensity

Add Light - will add a new light to the current scene.

Delete - will delete the current light from the scene.

Procedure:

1. Select Lights from the tools menu.

2. Set X, Y, Z, Intensity, SoftSize of the current light.

3. Press "> >" to change settings for particular lights. Use "Add Light" or "Delete" to change number of lights.

4. Select "Exit Lights" and "Render" to see if the current light setting illuminates the scene in a satisfactory way.

Example:

- Select the "Dice" from the choice of scenes using the "Load Workspace" option on the main menu. Enter Scene Composition. When the scene appears select Render.

- Observe the rendered scene. Double click the right mouse button and select "Exit Render" from the menu. From the wireframe representation, select Tools, then Lights.

- Use > > to select light #1. Select Delete, exit the Lights menu and select Render. Observe how the lighting of the scene has changed.

Purpose:

To load new objects into the current scene.

Procedure:

1. Select "AddObj" from the Base menu.

2. Select an object, use the "AS" option to rename it (optional), then select Load.

4. The newly-loaded object appears in the scene.

5. Select Done to get back to the Navigation menu.

Note:

The AddObj menu is similar to the load menu in Object Design. All the objects for the current scene must be in the current directory.

Purpose:

To save the current scene or load a new one.

Procedure:

1. Select "File" from the Base menu. File menu will appear.

2. Select a name for the scene to Load/Save. This can be done by clicking on list of files or typing the name in the text field.

3. Select Load/Save. **Caligari** will load/save the scene and return to the Navigation menu.

Note:

Caligari reminds the user to save the current scene whenever he exits the Scene Composition and the scene has been changed.

3D

QRENDER

In Scene Composition, a solid view of the scene can be obtained at any time by selecting QRender from the Base menu. QRender stands for Quick Render and it is much simpler than the final Broadcast Render which is covered in separate section.

Rendering capabilities in Scene Composition are more extensive than in the Object Design module. While most of the time **Caligari** displays the scene in wireframe form to achieve real time response, it is the rendered image of the scene which will be more impressive.

The key to impressive looking images is careful positioning of lights, economical use of available palettes and a balanced color scheme. Use of diffuse and specular coefficients assigned to various objects will also affect the final results. For example, a sphere colored red will look quite different when rendered with low diffuse coefficient and specular highlight, then rendered with a high diffuse coefficient and no specular highlight.

For the first rendering of a scene, **Caligari** needs to do computations to create the data necessary for solid rendering of a scene. After that, less computation are needed and the user can switch quickly between solid view and wireframe representation.

When "Render" is selected, the screen changes to solid view. All menus are eliminated to provide an unobstructed view of the object. The user can exit this view by **clicking the right button on the mouse**. This will bring up five buttons on the bottom of the screen: **Print Image, Save Image, Color, Setup** and **Exit Render.**

The user can now print the image on the printer, save it as a IFF file, change the color of any object, change the resolution or switch back to the wireframe view just like in Object Design.

3D(1)

COLOR

Color attributes are assigned to each object in the Object Design module. In Scene Composition, the color attributes cannot be changed. Their appearance however, is fully controllable in the scene.

For example, a car will be given color attribute one for the wheels and color attribute two for the body in the Object Design module. In the Scene Composition module the user can load the same car under three different names. Then, he can set the appearance of the color attribute one to gray and color attribute two to light blue for car1. After that, he can set the color attribute two to dark red for car2 and shiny blue for car3. Thus, all the cars will have grey wheels but the first will have a light blue body, the second will have a dark red body and the third car's body will be blue and shiny.

There are three appearance attributes that can be assigned to each color attribute for a selected object: hue (**Color**), intensity (**Diffuseness**) and shininess (**Specularity**).

Hue (color) can be assigned to the object simply by selecting one of the color bands (palettes). For a selected hue (color band), intensity and shininess can be specified with horizontal sliders for diffuseness and specularity.

Just as in Object Design, the number of palettes, as well as the number of shades for each object, can be easily changed. In addition, Hue, Saturation and Value sliders allow the user harmonize the color scheme. All of these settings are part of the scene and will be saved together with it. Typically, on a new project, setting up a color scheme for the scene will be one of the user's first decisions (although most users will prefer to this in Brender section).

Example:

- Create a simple scene with a few spheres in it and render it. Select "Color" to enter the Color menu.

- Select one of the color bands above the menu. The H, S, and V sliders can now be used to interactively adjust, in real time, Hue, Saturation and Intensity of the selected palette. Do this for all bands including the background band (on the left) to achieve a pleasing combination of colors. Notice, that when the band with the color in which the spheres are rendered is modified, the spheres also change appearance.

- Select one of the bands (not the rightmost band), press down right mouse button and move the mouse. The band will grow or shrink to include more or fewer shades. If more shades for the palette are chosen, the other bands will shrink proportionally.

Purpose:

To specify the hue, intensity and shininess of a color attribute of an object.

Procedure:

1. Press right button in solid view mode to bring up the menu. Select "Color" from the menu.

2. Select an object from the scene. The rest of the scene will become shaded out.

3. Select Material# (the color attribute) of the current object to be changed by typing from the keyboard.

4. Set "Diffuse" and "Specular" coefficients. Select active color from the palettes above the menu.

5. Depress "Set Material" to assign new values to the attribute. If you double click on the button, the object will be also rerendered.

Note:

All palette functions are also available. You can increase the number of colors, and change the hue, saturation, and value of any color (palette) interactively. It is important to remember how many color attributes were assigned to an object in Object Design. Only these attributes can be manipulated in the Scene Composition module.

Example:

- Create a simple scene with a few spheres in it and render it. Select Color to enter the Color menu.

- Select one of the spheres. Set the Material # to 1. Set the Specular slider to the right and double click on "Set Material" to rerender the sphere. Now the sphere will have a highlight.

- Select another sphere. Set the Material # to 1 and select a different palette. Double click on "Set Material" and watch the sphere change color.

- Select yet another sphere and set the Material # to 1. Slide the Diffuse slider to the left. Double click on "Set Material" and watch as the sphere is rerendered in a darker color.

Purpose:

To change the resolution of the solid view display, to select overscan, to clear, to rerender or to display color bars.

Procedure:

1. Double click in solid view mode to bring up the menu. Select "Color" and "Setup" from the menu.

2. Select Hires/Lores or other options (see note).

3. Select Render and the scene will be rerendered with the new resolution values.

Note:

Hires/Lores switches alternatively between high and low resolution. Lower resolution is 364x482 pixels and high resolution (default) is 740x482 pixels.

In low resolution more colors are available.

Other options:

Underscan/Overscan/DpaintOverscan allows the user to set underscaned or overscaned image. Special option will assure the compatibility with Dpaint image format.

Clear/Rerender will clear the screen or rerender it again.

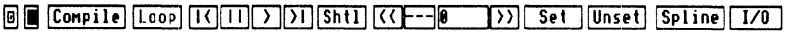
Color Bars will display standard NTSC color bars.

4. ANIMATION

4A. INTERACTIVE ANIMATION EDITING

Caligari supports a powerful interactive editing technique for scripts, one based on keyframes. Instead of manually writing the script for animation as in previous versions, the user can simply guide individual actors (objects to be animated) through a sequence of most important positions - keyframes. Intermediate frames are then automatically calculated with linear or spline based interpolation.

The user can enter the animation by selecting Anim button from the base strip in Scene module. Once selected, the animation tool bar appears with following buttons:



Compile

Compile can be selected when the user made changes to the animation and wants to preview them. Once compiled, the animation can be previewed in real time and further edited.

Program compiles animation structures residing in memory to preview structures, or if these structures do not exist, compiles script file previously loaded (default script file name "Default.script"). As an side effect, work script file CalScratch:scratch.script will be created.

After compilation program enters the preview mode (one bit plane screen). The user can jump to the scene design space at any time by selecting any button in navigation menu.

Loop

Toggle - toggles between preview mode Loop and One shot. If selected the preview will be played repeatedly.

"|<"

If there is a compiled script in the memory (created by compile), enters preview mode and jumps to the first frame of the animation.

"||"

Stops previewing. If in the Scene view - jump to preview mode.

">"

Enters preview mode Play (animation is previewed with maximum speed).

">|"

Jumps to the last frame of the animation. Enters preview mode, if it is possible.

Shtl

Enters preview mode **Shuttle** or **Jog**. By depressing the **left** mouse button the user can change the speed of preview interactively (in effect shuttle the tape). Moving the mouse left or right accelerates/ deccelerates the preview. If the preview is slowed down to a stop, further deccelaration will result in the movement in the opposite direction.

By depressing **both** mouse buttons at once the Jog mode is entered. In this mode the user drags with the mouse preview frames in any direction and in any speed (may be faster then maximum speed available by normal previewing).

"< <"

Step backward one frame. Does not work in preview mode. In Scene module the program computes desired frame and draws it. The user may than edit this frame.

MRS

This is a text field with the space for three characters and it is not editable. This field displays whether the current frame is a keyframe for the selected object and which action on the object has been defined for this frame. Three letters are possible - M for move, R - for rotation and S - for scaling. The starting frame for each object is always of the type "MRS".

Frame number

Displays the current frame number. This is automatically updated by preview operations. The user may enter any number into this field and the program will jump to the corresponding frame, whether the user is in the preview or the Scene design space.

"> >"

Step forward one frame. In the Scene design the software computes the next frame, which may be then edited. If no script exists, current frame in the scene is stored in frame 0 and the anim structures are automatically initialized.

Set

If depressed, the current scene position for the Object/Eye is stored as a keyframe. Affected are only the objects, **which were moved** (move/rotate/scale keyframe) with the desired type of operation. Thus the user can move the viewpoint around and look at the scene from different angles without saving the current viewpoint as a keyframe.

If the user wants to save the current frame as the keyframe for the eye position, EYE button in navigation must be selected when Set is pressed. The user must press Set before changing the frame number, otherwise the keyframe position will be lost.

Frames may be generated by a valid navigation operation (such as move, rotate or scale) with any object or eye. There are some restrictions:

- rotation angles must be less than 180 degrees (Caligari remembers only keyframe positions, not operations leading to the keyframe). If the user uses Rotate command in the script, internaly it will be represented as "WROTATE" command. This may lead to a difference in the position of rotated object when the preview is stopped and user attempts to render a particular frame.

- Caligari supports a large variety of rotation operations, which allow to refine very easy and desirable orientation of an object. But it is not possible to follow these rotations and calculate in-betweening orientation then. That is why Caligari simply interpolates between

orientations in key frames and rotates object the shortest way between them.

- objects must be scaled in object coordinate system. For initial frame however, the objects can be scaled arbitrarily.

If the user wants to begin object move at a certain frame, he must select this frame, select move, move object, undo the move and select Set. Otherwise the program would interpret the movement as starting from frame 0.

Unset

Deletes current keyframe. All the action defined for the keyframe will be lost.

If frame number is 0, Unset deletes the whole animation script. New objects can be loaded, new scene animated.

The user **must unset the frame 0 when he wants to exit animation**. It is not enough just to close the animation tool bar.

Spline

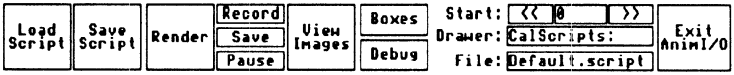
When set, positions between keyframes are computed by spline. This results in smoother motion of the objects and the viewpoint.

If Spline is unselected, the positions are computed by the linear interpolation. When the user is in the scene design space, position is updated immediately; when in the preview mode, animation must be recompiled before the user can see the updated animation.

I/O

This is the input/Output submenu

This is the submenu for animation input/output operations such as loading/saving of scripts and recording options for hard disk and VCR's. When selected, the following toolbar is displayed:



Load Script

Displays the file menu with scripts in the current directory. Program automatically inserts the extension if the user does not type it manually. Default name is Default.script.

Save Script

Displays the current directory with the existing scripts. The user can save an existing script, or type in a new name for the newly defined script. If there is no script in the memory, error message will be displayed.

Render

Renders animation (from frame given by Start gadget to the end) in full color instead of the wireframe form. This animation can be saved to the disk or recorded to the VCR.

Rendering is done in QRender, when invoked from scene design, or in BRender when invoked from broadcast renderer.

If no splines are used, script residing in the memory will be lost. To define the script again the user must recompile the animation.

Record

Toggle for recording the animation rendering to the video tape.

Save

Toggle for saving pictures from rendering to harddisk (in Caligari: directory).

Pause

Toggle for pausing after rendering each BRender frame.

Boxes

Toggle for previewing in full wireframe or bounding boxes only.

Debug

This option displays lines of script as it is compiled. It also displays syntax errors.

Start: < string gadget >

The user can specify which should be the starting frame of the animation. Animations structures are always created for full animation. It gives the beginning frame for ViewImages and Render too.

Drawer:

Displays the current directory. From here the path may be set directly.

File

Displays the current scriptfile. Extension .script is automatically added. The user can type the name here, and he does not need to use LoadScript function.

Exit AnimI/O

Return to animation menu.

ANIMATION SESSION

Typically, the user would design a scene for the starting configuration of the intended animation and save it. Then he would bring up the animation tool bar and type in the number of the frame he wants to edit (frame 0 will be defined automatically).

Then the user can manipulate the objects or the viewpoint to create the desired configuration for entered frame and press Set (remember that Set works separately for objects and the Eye). After the frame is set, it can be compiled and previewed or other key frames can be defined. The new key frame can be defined before or after the existing keyframe, even if the current keyframe is the last one in the animation.

It is a good idea to disable Dynamic Pick while in the animation section, to prevent undesired motion of wrong objects. It is also a good idea to preview the animation after even a small change in keyframes.

In the preview, JOG option is very valuable for fast and accurate selecting of the next frame to be edited as the user can visually select the part of an animation where the edit is most needed.

After the script is saved, it can be further manipulated in the text editor to add special effects such as acceleration, or precise positioning with numerical input. The user can move between interactive and text based editing, except for a few commands which must be added after the interactive edit.

4B. SCRIPT BASED ANIMATION

Originally the only option for creating scripts, it is a valuable tool for creating special effects.

The script can be written on any Amiga text editor. Each line of the script contains one animation command. Typically, the script will consist of a few "paragraphs" - blocks of commands, each starting with a GO command followed by set of commands to be executed in particular frame range (as defined by the previous GO). After the script is written, named and saved, it can be loaded into Caligari.

Following simple script will create two seconds of animation. In this script car will be translated relatively from its current position 50 units along the Y axis (perhaps along the street) and base will be rotated 360 degrees. While this is happening camera closes up on the scene.

```
GO 60

MOVE car 0 50 0

ROTATE base 0 0 360

MOVE EYE 0 0 100
```

The Caligari script animator is case insensitive to commands and to the EYE, but it is case sensitive to the names of objects.

The semi-colon, ";", may be used anywhere on a script line to indicate that the remainder of the line is a comment or is to be ignored. Blank lines are also ignored.

Caligari executes a "GO 1" command automatically at the beginning of each script. At the end of each script a "PURGE" command is also automatically executed.

SCRIPT SYNTAX

MOVE

Purpose:

To move an object or eye relative to its current position.

Syntax:

MOVE {object | EYE } x y z [t]

Options:

object - name of the object to be translated

EYE - eye will be translated

x, y, z - spatial increments to be added to current position

t - number of frames (optional)

Example:

MOVE car 0 100 0 20

MOVE EYE 20 30 10

Description:

The Move command will translate the specified object or the eye relative to its current position over the number of frames specified in the "t" parameter. If the frame count is not specified, then the MOVE will be executed over number of frames specified in the preceding GO command.

AMOVE

Purpose:

To move an object or the eye absolutely to a given point in space.

Syntax:

AMOVE {object | EYE } x y z [t]

Options:

object - name of the object to be translated

EYE - eye will be translated

x, y, z - world coordinates where the object or eye will be moved

t - number of frames (optional)

Example:

AMOVE car 110 100 0

AMOVE EYE 0 0 200 10

Description:

Absolute move will translate the specified object or eye from its current position over the number of frames specified in the "t" parameter to the world coordinates specified. If the frame count is not specified, then AMOVE will be executed over the number of frames specified in the preceding GO command.

SCALE

Purpose:

To change the size of an object.

Syntax:

SCALE object x y z [t]

Options:

object - name of the object to be scaled

x, y, z - size increments

t - number of frames (optional)

Example:

SCALE car 1 1 10 20

Description:

Scale command will scale an object over a number of frames. If the frame count is not specified, then SCALE will be executed over the number of frames specified in the preceding GO command.

ROTATE

Purpose:

To rotate an object or eye relatively.

Syntax:

ROTATE {object | EYE } x y z [t]

Options:

object - name of the object to be rotated

EYE - eye will be rotated

x, y, z - angular increments for rotation

t - number of frames (optional)

Example:

ROTATE car 0 90 0 20

ROTATE EYE 720 0 0

Description:

If more than 360 degree increment is specified, object will rotate around the specified axis more than one time. However this information will be lost once the user edit the object interactively since "rotate" will revert to "WROTATE" command.

If the frame count is not specified, then ROTATE will be executed over number of frames specified in the preceding GO command.

AROTATE

Purpose:

To rotate an object or eye around a specified axis.

Syntax:

AROTATE {object | EYE } angle x1 y1 z1 x2 y2 z2 [t]

Options:

object - name of the object to be rotated

EYE - eye will be rotated

x1, y1, z1 - first point defining the rotational axis

x2, y2, z2 - second point defining the rotational axis

t - number of frames (optional)

Example:

AROTATE car 360 0 90 0 20 0 0

Description:

This command enables objects to orbit on arc in 3D space. If the frame count is not specified, then AROTATE will be executed over number of frames specified in the preceding GO command. Again the angular value may be lost once the user edit the object interactively since "Arotate" will revert to "AMOVE" and "WROTATE" commands.

Purpose:

To enable the eye to follow an object.

Syntax:

LOOKAT object [t]

Options:

object - name of the object for eye to follow

t - number of frames (optional)

Example:

LOOKAT car 20

Note:

This command enables to steer the eye to follow an object. It is particularly useful with invisible objects.

INVISIBLE

Purpose:

To make an object disappear for part of the animation.

Syntax:

INVISIBLE object

Options:

object - name of the object to be made invisible

Example:

INVISIBLE car

Description:

A small tetrahedron can be made invisible and then LOOKAT command can guide the motion of an eye. Alternating visibility of objects under the script control can create interesting animation effects. (See VISIBLE)

VISIBLE

Purpose:

To make an object appear again.

Syntax:

VISIBLE object

Options:

object - name of the object to be made visible

Example:

VISIBLE car

Description:

Objects are visible as a default. VISIBLE command is useful for special effects. (See INVISIBLE)

PAUSE

Purpose:

To do nothing for a few frames.

Syntax:

PAUSE t

Options:

t - number of frames

Example:

PAUSE 20

Description:

Scene will stay unchanged for a number of frames.

SKIP

Purpose:

To speed up editing process by not previewing already finished parts of script.

Syntax:

SKIP t

Options:

t - number of frames

Example:

SKIP 45

Note:

Program will execute all commands specified in the script but will not render frames affected by Skip command.

Purpose:

To execute next few frames of a script.

Syntax:

GO t

Options:

t - number of frames

Example:

GO 60

Note:

GO works in prefix notation, i.e. it effects commands after its occurrence. Also, GO specifies the default number of frames for all subsequent commands with optional "t" parameters.

Purpose:

To change overall speed of script execution.

Syntax:

RATE F

Options:

F - a floating point factor by which to scale time.

Example:

RATE 2

Will create half as many frames for coarser but faster motion.

RATE .5

Will create twice as many frames for finer and slower motion.

Purpose:

To clear all ongoing actions in the script.

Syntax:

PURGE

Description:

This command will prevent all unexpired commands from executing.

;

Purpose:

To insert comments into the script.

Example:

```
MOVE EYE 0 0 10   ; end of first section

; Car starts to accelerate

MOVE car 0 20 0 10
```

Note:

Comments can be placed anywhere after the ";" symbol

New additions

When the user is not using the interactive editor capabilities, script file language is fully compatible with the old version of Caligari.

But there are some restrictions when switching between interactive and text editing:

- MOVE command is always changed to AMOVE

- ROTATE is changed to WROTATE (see later).

Angles greater (in absolute value, however) than 180 degrees are decreased.

- AROTATE is changed to WROTATE and AMOVE - thus arc of circle is changed to line.

- LOOKAT is changed to WROTATE EYE (LOOKAT remains).

Do not use RATE and PURGE commands, if you wish to edit script interactively.

All script commands are treated in the old manner, when compiled first time and valid preview structures are created. Only animation structures have these restrictions and thus animation saved creates simply another scriptfile. Second compile may give different results (after some editing activities).

Purpose:

Substitution for rotation functions in interactive mode.

Syntax:

WROTATE object/EYE angle x y z [frames]

General substitute for the rotation functions. Rotates an object or the Eye along an explicitly specified axis by an specified angle. Rotation is always made with relation to the object or Eye's center. The axis is defined by X, Y, Z point in space defining vector passing through 0, 0, 0 (center of the world coordinate space).

Rotation angle must be less than 180 degrees in absolute value.

example

WROTATE object 45. 0. 0. 1.

This will result in a 45 degree rotation about the world Z axis.

Purpose:

To slow down or speed up the motion of an object or the viewpoint.

Syntax

ACCELERATE object/EYE move_flags acceleration

This command may be added to the script file only by an external text editor. It is used to create more realistic motion of objects. Velocity between two keyframes is not constant, but it is accelerated or deccelerated.

object_name/EYEis a string with the name of the object to be accelerated (or the EYE)

move_flags is a string which describes types of move to be affected. Characters in this string can be (case and order is not sensitive):

m - move

r - rotation

s - scale for the objects, zoom for eye

accelerationis floating point number from interval 5.0, 5.0. Negative values represent deceleration. Value of 0.0 represents a normal move without acceleration.

User must enter this command in script file after the move specificationsand in same go sequence.

Examples:

GO 15

MOVE cube x y z 10

SCALE cube x y z

ACCELERATE cube ts 2.3

will accelerate cube in translation in next 10 frames and in scaling in next 15 frames.

GO 15

MOVE cube x y z 10

ACCELERATE cube ts 2.3

SCALE cube x y z

will accelerate cube only in translation, and finally

GO 15

MOVE cube x y z 10

SCALE cube x y z

GO 1

ACCELERATE cube ts 2.3

will not accelerate object cube at all.

Example

There are two objects in the scene - a cube and a cone.

To move the cube from initial position to new position from begin of animation to its end, rotate it from frame 10 to frame 20 and scale cone from frame 5 to 15 the following needs to be done:

- set initial position of the scene and save it

- jump to frame 5 (by < or > or string gadget)

- scale cone and undo operation (scaling is from frame 5, not 0)

- set

- jump to the frame 10

- rotate cube and undo

- set

- jump to frame 15

- set final scaling for cone

- set

- jump to frame 20

- set final rotation for cube

- set

- jump to ending frame (say 30)

- set final position of cube

- set

SaveScript gives the following script file:

GO 5

AMOVE cube x y z 30

GO 5

SCALE cone x y z 10

GO 5

WROTATE cube angle x y z 10

GO 15

Now you can add (using a text editor) commands such as ACCELERATE cube t 5 (for moving cube from zero velocity to maximum speed in frame 30) or MOVELIGHT or SETATTRIB and so on to design the complete animation with light and color effects.

These steps can be mixed in any order; only the first step must be first (set initial scene) always.

Purpose

To animate the lights.

Syntax:

[A]MOVELIGHT lgtnum x y z [t]

where:

lgtnum = number of the light to move

x, y, z = coordinates or delta values depending on AMOVE/MOVE

t = optional number of frames

MOVELIGHT will move a light a relative distance in X, Y, Z, while AMOVELIGHT moves the light to new absolute world coordinates. For both commands, t is an optional parameter specifying the number of frames over which to move the light. These commands work in both Brend, and Qrend. In wireframe previews, the lights are moved, but do not affect the wireframe output.

Example:

MOVELIGHT 1 0 0 10 50

This will move light 1 10 units above the center of the perspective reference grid through 50 frames.

Purpose:

To animate spotlight.

Syntax:

LIGHTOBJECT object lgtnum t

object - name of object to point light at

lgtnum - number of light

t - frames

LIGHTOBJECT is used to direct a spotlight at a particular object. If t is specified, the light will be directed for t frames, and if t is not specified, then it will be directed at the object for the duration of the current GO sequence. This command only works in BRender animations and only with spotlights.

Example:

```
LIGHTOBJECT NoName 1 5
```

This will direct light 1 at object NoName for 5 frames.

Purpose:

To animate the environment.

Syntax

SETENV objname matnum envname [t]

RENDERENV objname matnum envname [t]

where:

objname = name of object

matnum = object material number

envname = name of env to load/render

t = frame to make env change or number of frames to render new env

SETENV will load a new environment for the specified object. The envname parameter is the name of the environment to load. The environment MUST be in the CalEnviron: directory (the normal place for such things). If no t parameter is specified, the environment is set when the next frame is rendered. If t is specified, Then the environment will be changed at frame t relative to the current GO sequence (i.e. if the current GO renders frames 35-50 of the animation, the environment will be changed at frame 35 + t). RENDERENV will render a new 2D environment for the specified object for a number of frames. The envname parameter specifies the name of the environment that will be rendered. The environment will be in the CalEnviron: directory. The optional t parameter specifies the number of frames for which the environment should be rendered. If no t is specified, it defaults to the number of the current GO sequence.

When animating, if the current render type is not BRender, these commands will be ignored.

Examples:

SETENV NoName 1 sky.e2d 43

This will set the environment map of material 1 of object NoName to sky.e2d at frame 43 of the current GO sequence.

RENDERENV NoName 3 scene.e2d 6

This will render the environment map scene.e2d and apply it to material 3 of object NoName through 6 frames.

Purpose:

To animate texture maps.

Syntax

SETTEXTURE object matnum file [t]

object - name of object

matnum - material number

file - name of texture

t - frame at which to change the texture

SETTEXTURE will load a new texture for the specified object and material at the frame t. This allows you to animate textures. It is not full rotoscoping. The texture must be in the CalTexture: directory. If no t parameter is specified, the texture is set when the next frame is rendered. If t is specified, Then the texture will be changed at frame t relative to the current GO sequence (i.e. if the current GO renders frames 35-50 of the animation, the texture will be changed at frame 35 + t). This command only works when doing Brender animations. It will be ignored in previews and Quick Render animations.

Example:

SETTEXTURE NoName 1 clouds.txr 5

This will set the texture of material 1 of object NoName to clouds.txr at frame 5 of the current GO sequence.

Purpose

SETATTRIBUTES is for animating material attributes.

Syntax

SETATTRIBUTES object matnum {rgb | hsv} {gradual | immediate} {r | h} {g | s} {b | v} [trans] [ambi] [diff] [shin] [spec] [fangle] [t]

object - name of object

matnum - material number

{rgb | hsv} - how you have specified the color data that follows rgb for RGB data, or hsv for HSV data {gradual | immediate} - how to make the transition to these new attributes: an immediate change at frame t or a gradual change over t frames

{r | h} - Red or Hue data

{g | s} - Green or Saturation data

{b | v} - Blue or Value data

trans - transparency

ambi - ambience

diff - diffusion

shin - specular power coefficient

spec - specularity

fangle - angle for autofacet shading

t - frame number

The syntax is more complicated than most other script commands, but it gives the user great flexibility. The parameters up to and in-

cluding {b | v} are required. The rest are optional. The {rgb | hsv} parameter allows you to specify color information in whichever color space you feel most comfortable. RGB data goes from 0 to 255. HSV data goes from 0.0 to 1.0 except Hue which goes from -2.0 to +2.0.

The {gradual | immediate} parameter allows you to specify the type of transition to make from one set of attributes to another, and changes the interpretation of the t parameter. If gradual is specified, the change will be progressive over the time of the current GO sequence if t is not specified, and if t is specified, then over t frames. If immediate is specified, then the material attributes will change immediately at frame t, and if t is not specified, the attributes are set when the next frame is rendered.

The optional trans parameter is for setting transparency and is in the range 0 to 255. The optional ambi parameter is for setting ambience and is in the range 0.0 to 1.0. The optional diff parameter is for setting diffusion and is in the range 0.0 to 1.0. The ambience and diffusion parameters are checked against each other and if their total is not 1.0 or less, one of them will be clipped. The optional shin and spec parameter are for setting specularity and they are in the range 0.0 to 1.0. This command works in both BRender and Quick Render animations.

Examples:

SETATTRIBUTES NoName 1 rgb gradual 255 128 0 50

This will gradually change material 1 of object NoName to a transparent orange through the number of frames of the current GO sequence.

SETATTRIBUTES NoName 3 hsv immed 0 1.0 1.0 255 .5 .25 .75 36

This will immediately change material 3 of object NoName to opaque cyan at frame 35 of the current GO sequence.